TU Braunschweig – Niedersächsisches Forschungszentrum für Luftfahrt

Berichte aus der Luft- und Raumfahrttechnik

Forschungsbericht 2017-25

Aerodynamic Optimization of Helicopter Rotor Blades using Variable Fidelity Methods

Gunther Andreas Wilke

Deutsches Zentrum für Luft- und Raumfahrt
Institut für Aerodynamik und Strömungstechnik
Braunschweig

Diese Veröffentlichung wird gleichzeitig in der Berichtsreihe „NFL - Forschungsberichte“ geführt.

Diese Arbeit erscheint gleichzeitig als von der Fakultät für Maschinenbau der Technischen Universität Carolo-Wilhelmina zu Braunschweig zur Erlangung des akademischen Grades eines Doktor-Ingenieurs genehmigte Dissertation.

Aerodynamic Optimization of Helicopter Rotor Blades using Variable Fidelity Methods

Von der Fakultät für Maschinenbau
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung der Würde

eines Doktor-Ingenieurs (Dr.-Ing.)

genehmigte Dissertation

von: Gunther Andreas Wilke
geboren in: Marl, Deutschland

eingereicht am: 16. März 2017
mündliche Prüfung am: 28. November 2017

Vorsitz: Prof. Dr.-Ing. Jens Friedrichs
Gutachter: Prof. Dr.-Ing. Cord-Christian Rossow
Prof. Prof. Dr.-Ing. Rolf Radespiel

2017

Abstract

The motivation of this work is to reduce the overall required computational resources of the aerodynamic optimization process of helicopter rotor blades. Therefore an approach already applied to fixed wing aircraft is now ported to the helicopter rotor blade optimization. It is a surrogate based optimization process including variable-fidelity methods.

After setting up the framework to accomplish this task, a theoretical and practical investigation of various aerodynamic models ranging from the blade element theory to computational fluid dynamics (CFD) for helicopter rotors is performed. This is done with the 7A model rotor in two flight conditions, namely hover and forward flight. From this examination a potential subset of methods is selected and further analyzed in applied variable-fidelity optimizations.

These optimizations are benchmarked with a state of the art single-fidelity process. Potential resource reductions of up to 85 % in computing costs are observed and at the same time unfeasible model combinations are identified. The benefit is demonstrated for two underlying rotor configurations, the 7A model rotor and the reference rotor of the European CleanSky Green Rotorcraft (GRC) research project. While the 7A rotor is parameterized with four design variables, the GRC rotor optimization is performed at a more industrial relevant level with ten design variables and two structural constraints.

The final selection of model combinations is the blade element theory enhanced with a prescribed wake model and a multi-bladed chimera setup solving the RANS equations in forward flight, while for hover the solution of the inviscid Euler equations supply the trend for the solution of the RANS equations on a periodic mesh.

The employment of the variable-fidelity approach to multi-objective scenarios proves to be highly beneficial and the need to perform multi-objective optimizations for rotor blade design is highlighted. The performed optimizations in this work provide a selection of potential future blade designs.

Zusammenfassung

Die Motivation dieser Arbeit ist es, die insgesamt erforderlichen Berechnungsressourcen des aerodynamischen Optimierungsprozesses von Hubschrauber-Rotorblättern zu reduzieren. Daher wird ein bereits auf Starrflügler angewandter Ansatz nun auf die Helikopter-Rotorblatt-Optimierung portiert. Es handelt sich um einen Ersatzmodell-basierten Optimierungsprozess mit Methoden unterschiedlicher physikalischer Modellierungstiefe.

Nach der Erstellung einer Optimierungs-Umgebung für den Ersatzmodell-basierten Optimierungs-prozess mit Methoden unterschiedlicher Modellierungstiefe wird eine theoretische und praktische Untersuchung verschiedener aerodynamischer Modelle durchgeführt, die von der Blattelementtheorie bis zur numerischen Strömungsmechanik (CFD) für Hubschrauberrotoren reichen. Dies geschieht mit dem 7A-Modell Rotor in zwei Flugbedingungen, nämlich dem Schwebe- und Vorwärtsflug. Aus dieser Untersuchung wird eine mögliche Teilmenge von Methoden ausgewählt und im Ersatzmodell-basierten Optimierungsprozess mit Methoden unterschiedlicher Modellierungstiefe angewandt und weiter analysiert.

Die Optimierungen, welche die Modelle mit variabler physikalischer Eindringtiefe nutzen, werden mit Optimierungen, welche nur eine physikalische Tiefe nutzen, verglichen. Potenzielle Ressourcenreduktionen von bis zu 85 % der Rechenkosten werden beobachtet und gleichzeitig werden ungünstige Modellkombinationen identifiziert. Der Nutzen dieses Vorgehen wird für zwei zugrunde liegende Rotor-Konfigurationen, den 7A-Modell-Rotor und den Referenz-Rotor des europäischen CleanSky Green Rotorcraft (GRC) Forschungsprojektes, demonstriert. Während der 7A-Rotor mit vier Entwurfsvariablen parametrisiert wird, wird die GRC-Rotoroptimierung auf einer industriell relevanteren Ebene mit zehn Entwurfsvariablen und zwei strukturellen Nebenbedingungen durchgeführt.

Die endgültige Modellkombination im Vorwärtsflug ist die Blattelementtheorie, welche mit einem vorgeschriebenen Wirbel-Modell erweitert wird, und die numerische Strömungslösung der RANS-Gleichungen, welche auf einem Mehr-Blatt-Chimären-Setup gelöst werden. Als Modell- kombination für den Schwebeflug, werden die reibungsfreien Euler-Gleichungen mit den RANS- Gleichungen auf einem periodischen Netz genutzt.

Den Ansatz von Methoden unterschiedlicher physikalischer Modellierungstiefe für die Optimierungen mit mehreren Zielfunktionen zu nutzen erweist sich als sehr vorteilhaft und die Notwendigkeit, Mehrpunkt-Optimierungen für den Rotorblatt-Entwurf durchzuführen, wird hervorgehoben. Die durchgeführten Optimierungen in dieser Arbeit bringen eine Auswahl an möglichen zukünftigen Rotorblatt-Entwürfen hervor.

Acknowledgements

During the preparation of this thesis I learned a lot about helicopter aerodynamics and surrogate models. But not only on a technical and scientific side I learned a lot, also interacting and communicating with people, their different viewpoints about things, have contributed to my overall positive experience during this work.

I am thankful to my doctoral father, Professor Cord-Christian Rossow for supervising my thesis as well as to Professor Rolf Radespiel as the co-examiner and Professor Jens Friedrichs for chairing the PhD committee. I am also thankful to my former branch head, Professor Norbert Kroll for organizing the initial funding for this thesis as well as giving me the freedom and liberty to excel in my work. I am also grateful that he granted me the time to visit the graduate program from DLR. I am also thankful that my current branch head Doctor Thorsten Schwarz to advise me during the thesis and enable me to visit the European Rotorcraft Forum, which has been a very beneficial source of exchange.

The enrollment in the Green Rotorcraft Program after my official PhD contract ran out was a blessing. Thanks to Stefanie Link, Biel Ortun and Simon Spurway, I was given the opportunity to test out my 'fledging' framework for real rotorcraft applications and discuss the less and more successful rotor designs. The research leading to the results related to the GRC rotor has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) for the Clean Sky Joint Technology Initiative under grant agreement n^o CSJU-GAM-GRC-2008-001.

I am especially thankful to all the colleagues at the Institute of Aerodynamics and Flow Technology, who were always available for questions and had an open mind in supporting me. Big thanks to Manfred Imiela, whose work I continued and who has guided me during most my endeavors and setting an excellent starting point for my work. Thank you to Ralf Zimmermann, Chunna Li, Zhong-Hua Han and Stefan Görtz for the numerous discussions on surrogate models and surrogate based optimization. I am grateful for Jianping Yin's support on the unsteady panel method, and his insight to further fine tune it for my applications. Thank you to Jochen Raddatz for sharing all his FLOWer experience and knowledge with me. A special thanks to Christoph Heister for his 'external view' as well as his clever ideas on presenting things. Thank you to Gerrit Heilers for preparing his Master thesis under my supervision. I learned a lot during this period. Without him, the idea of the crashmap may have never come to my mind. I would also like to acknowledge my gratitude to Caslav Ilic for all his ideas on python programming and general thoughts on optimizations.

Last, but not least, I am thankful for my parents, who believed in me throughout the preparation of this thesis and taught me to keep all my documents together. Also, special thanks to my wife, Ann-Katrin and my daughter Adela, who had to bear with me during the final stages.

List of Preceding Publications

During the preparation of this thesis, multiple papers have been written for the European Rotorcraft Forum. They fully or partially contain work related to this thesis and are explained in detail below:

- [1] G. Wilke: Variable Fidelity Optimization of Required Power of Rotor Blades: Investigation of Aerodynamic Models and their Application, *38th European Rotorcraft Forum*, 2012: This publication contains a summary of the sensitivity study shown in Section 3.3 and first application results of the variable-fidelity method listed in Section 4.1.3

- [2] G. Wilke: Multi-Objective Optimizations in Rotor Aerodynamics using Variable Fidelity Simulations, *39th European Rotorcraft Forum*, 2013: In this paper, the application of the multi-objective optimization strategy in connection with the variable-fidelity approach as formulated in Section 2.4.2 is demonstrated for the 7AD rotor and shows the preliminary version of the results in Section 4.1.3 and Section 4.1.4

- [3] M. Imiela and G. Wilke: Passive Blade Optimization and Evaluation in Off-Design Conditions, *39th European Rotorcraft Forum*, 2013: Contributes to the research related to the Green Rotorcraft (GRC) Program in Section 4.2 featuring a mid-fidelity forward flight optimized blade with nine design variables, which is the contribution of the Wilke to this paper.

- [4] G. Wilke: Applying Multi-Objective Variable-Fidelity Optimization Techniques to Industrial Scale Rotors: Blade Designs for CleanSky, *41th European Rotorcraft Forum*, 2015: Here, the high-fidelity multi-objective optimization of the GRC rotor are presented along with an acoustic evalution of these blades. The latter is not part of the here presented research, however, the optimization results are also shown in Section 4.2.5

Contents

List of Figures

List of Tables

Nomenclature

Acronyms, Surrogate Based Optimization Context

CVT	Central Voronoi Tessellation
DE	Differential Evolutionary algorithm
DEGL	Differential Evolutionary algorithm with the Global Local extension by Das et al. [133]
DoE	Design of Experiments
FF	Full factorial design
HF	High-fidelity based data or surrogate function
HFM	High-fidelity method, method at high level of fidelity
LF	Low-Fidelity based data or surrogate function
LFM	Low-fidelity method, method at very low level of fidelity
LHS	Latin Hypercube Sampling
MF	Mid-fidelity based data or surrogate function
MFM	Mid-fidelity method, method at intermediate level of fidelity
MMC	Morris & Mitchel Criterion, a metric to quantify the quality of a DoE.
QMC	Quasi Monte-Carlo sequences
RBF	Radial Basis Function
SBO	Surrogate Based Optimization
SF	Single-fidelity, here referred to cases which only use one fidelity level.

VF	Variable-fidelity, here referred to cases where multiple fidelity levels are applied.
VFM	Variable-fidelity methods or models

Acronyms, Aerodynamic Modeling Context

BEMT	Blade Element Momentum Theory
BET	Blade Element Theory
EU	CFD solution from inviscid Euler equations on coarse meshes and single blade model
FISUW	The BET enhanced with a **FI**nite **S**tate **U**nsteady **W**ake inflow model
FNS	CFD solution from viscous RANS equations on fine meshes and multi-blade model
F.WAKE	The BET enhanced with a **free** wake model.
NS	CFD solution from viscous RANS equations on coarse meshes and single blade model
P.WAKE	The BET enhanced with a **prescribed** wake model.
UPM	Panel method including an unsteady free wake model
CFD	Computational fluid dynamics
RANS	Reynold-Averaged Navier-Stokes equations
RPM	Rounds/Revolutions per minute
TSP	Transonic small perturbation theory

Roman Letters, Surrogate Based Optimization Context

$\hat{s}(\vec{x})$	Kriging model error estimator
$\hat{y}(\vec{x})$	General surrogate prediction at $\vec{x}$
$\mathbf{C}$	Co-Kriging correlation matrix
$\mathbf{F}(\mathbf{X}_s)$	Regression matrix of sample points
$\mathbf{X}_s$	Matrix of samples points

$\vec{f}(\vec{x})$	Regression vector at point $\vec{x}$
$\vec{L}$,$\vec{G}$	Intermediate local and global individuals of the 'Differential Evolutionary Global and Local' algorithm
$\vec{v}$	Individual of the differential evolutionary algorithm
$\vec{W}$	Weights for radial basis or correlation functions
$\vec{x}$	Design vector or parameters
$\vec{Y_s}$	Responses of simulation put in a vector
c	Arbitrary constant
d	Distance between points in a hypercube
$DC(\vec{x})$	Design Confidence at point $\vec{x}$
$E[I(\vec{x}) \cap g(\vec{x}) < g_{max}]$	Constrained expected improvement at point $\vec{x}$
$E[I(\vec{x})]$	Expected improvement at point $\vec{x}$
$erf()$	Error function
F	Mutation rate of differential evolutionary algorithm
$f_{trend}(\vec{x})$	Kriging trend function at point $\vec{x}$
$g(\vec{x})$	Constraint function at point $\vec{x}$
g	Number of generations in an evolutionary process
H	Allowed tolerance for the Design or Pareto Confidence
J	Number of occurrences of same distances in a hypercube
k	Current spatial dimension
L	Maximum likelihood function
n	Number of samples
p	Kriging hyper parameter for the exponent of the radial basis function
$P[I(\vec{x})]$	Probability of Improvement at point $\vec{x}$
$PC(\mathbf{X}_{PF})$	Pareto Confidence for Pareto optimal points $\mathbf{X}_{PF}$

q	Exponential weight of Morris & Mitchell criterion
w	Weighting function to switch between global and local optimization of the 'Differential Evolutionary Global and Local' algorithm
$y(\vec{x})$	Goal or simulated function of $\vec{x}$

Roman Letters, Aerodynamic and Physical Modeling Context

Q_i	Kinetic energy
q_i	Generalized coordinates
T	Kinetic energy
U_{el}	Internal elastic energy

Greek Letters, Surrogate Based Optimization Context

$\alpha, \gamma, \rho, \sigma$	reflection, expansion, contraction and reduction coefficients for the simplex algorithm
δ	Additive constant for 'crashmap'
$\epsilon_{rbf}(\vec{x})$	Kriging error predictor at point $\vec{x}$
$\gamma(\vec{x})$	Additive bridge
$\hat{\sigma}$	Estimated variance of Gaussian process/Kriging model
Λ	Noise constant for Kriging and radial basis function approximations
$\Phi_p(\mathbf{X})$	Morris & Mitchell criterion for hypercube $\mathbf{X}$
$\rho(\vec{x})$	Multiplicative bridge or trend scaling factor
θ	Kriging hyper parameter for the width of the radial basis function
$\mathbf{\Psi}(\mathbf{X}_s)$	Correlation matrix of sample points $\vec{x}$
$\vec{\beta}$	Polynomial coefficients in vector format
$\vec{\psi}(\vec{x})$	Radial basis functions or correlation vector at point $\vec{x}$

Greek Letters, Aerodynamic Modeling Context

$0, c, e, r$	point corresponding to the center, contraction, expansion or reflection of the simplex algorithm

Subscripts and Indices, Surrogate Based Optimization Context

d	Number of design variables
i, j	Index counters
l	Index of goal functions
PF	Subscript for Pareto fronts
s	Subscript denoting a set of samples
v	Subscript denoting a vector
hfm	Index for data of the high-fidelity model
lfm	Index for data of the low-fidelity model
mfm	Index for data of the mid-fidelity model
vfm	Index for data of the variable-fidelity model

Subscripts and Indices, Physical Modeling Context

el	Subscript for elastic quantities

1 Introduction

1.1 Motivation

Helicopters play a very specific role in aviation. In contrast to airplanes they are inefficient in traveling long distances, but they have one key advantage over most regular aircraft. They can hover. And because they can hover, they can perform a multitude of tasks that regular aircraft cannot do. They require no landing strip and thus reach out and land in very confined areas. This includes bringing laborers to offshore sites, medical missions or reaching out to otherwise difficult to reach terrain. While these tasks are just a brief description of the many things a helicopter can achieve, helicopters also struggle with public as well as economical acceptance. One saying is: "Helicopters don't fly, they beat the air into submission", which may be very well the public perception. Helicopters consume a lot of fuel to operate in comparison with other transportation means, while featuring high noise levels at the same time and they need more maintenance than regular aircraft. In order to improve the overall approval of the helicopter, engineers are faced with a multitude of challenges. One key factor for designing a successful helicopter is a good rotor blade design.

The main rotor of the classical helicopter configuration fulfills two objectives. The first is to generate lift and thrust, the second objective is to steer the helicopter. Designing the blades for the main rotor is a highly multi-disciplinary task. In opposition to to fixed-wing aircraft, the blade rotates and thus encounters inertial forces that a wing usually does not. In particular the unsteady nature of most flight conditions, but also the flexible structure of the blades, additionally complicate the simulation process. Therefore, to follow the call for a greener rotorcraft is a demanding task and requires a lot of resources to further enhance the current designs. To alleviate the design process, numerical optimization strategies are more and more employed to automatically generate the best solution for a specific design question. The huge problem behind this is the tremendous amount of computational resources required to perform a high-fidelity aerodynamic simulation. An accurate simulation of a helicopter rotor blade takes up days or weeks on current computer clusters and the numerical optimization requires many of these simulations to arrive at the optimum.

This dissertation aims at accelerating the aerodynamic design process through numerical optimization of helicopter rotor blades by applying variable-fidelity surrogate models.

Four major aspects relevant for this study are identified; the rotor design problem, the simulation of physics, surrogate modelling and numerical optimization. The emphasis of this study is laid upon the simulation of physics with the particular focus on the aerodynamic modeling and surrogate models that accommodate multiple fidelities of these aerodynamic models.

1.2 State of the Art

The state of the art chapter is organized as follows; first the recent work done in generally optimizing rotor blades under different aspects is presented. Secondly, the analysis tools for these undertakings are introduced emphasizing the aerodynamic modeling options. On the other side are the available surrogate modelling techniques, which are reflected upon with respect to optimization problems. Afterwards, recently employed strategies for numerical aerodynamic optimization of rotor blades are discussed. Finally, the thesis and research goals of this work are outlined.

1.2.1 General Rotor Blade Optimization

In 1999 Celi [5] as well as Ganguli [6] in 2004 give good overviews of the advances in helicopter rotor optimization over the previous years. Celi states that the underlying physics of the analysis tool plays a key role in finding a robust and trustworthy optimum.

On the one hand, Ganguli claims that a local minimum is eventually better than a theoretical global optimum. His reasoning is that during a numerical optimization not all constraints can be included and certain design considerations are not taken into account. Thus, starting from an existing design and going to the next available minimum might be better in practice as it might feature a more trustworthy design than a not known or not understood globally optimal design. This leads to the remark by Ganguli that the local gradient based search algorithms might be more desirable as they are able to quickly find the next minimum in contrast to globally optimizing genetic algorithms. On the other hand, he states that the optimization of a rotor blade is clearly an aero-elastic problem, which is computationally expensive to solve and features multiple design solutions as well. This calls for a global optimization technique that does not rely on gradients, which are difficult to obtain for such a problem and have to be determined by finite differences. He advises to optimize rotors with a hybrid strategy, where a genetic algorithm or simulated annealing first coarsely finds the global optimum and then it is refined with gradient based methods. He also notes that the analysis tools are still fragile and the computation of finite differences can be erroneous as well as costly. An outlook is presented on the viability of surrogate models to decrease the numerical cost of optimizations with computational fluid dynamics (CFD). Proposed methods are polynomial response surfaces, Kriging, artificial neural networks or fuzzy logic systems. One key problem mentioned is the estimation of the accuracy of such models, but the advantage is the employment of resource intensive global optimization strategies.

Besides numerical optimization, often conceptual considerations and handcrafted rotor designs are the focal point of research. A good overview paper of helicopter rotor blade tips is presented by Brocklehurst and Barakos [7]. They debate about the merits of specific geometric modifications such as tapering and sweeping the blade tip as well as the employment of anhedral. They also list recent results in numerical helicopter simulations. Bebesel et al. [8] design a novel blade for the BK117 helicopter including practical considerations. They also perform flight tests proving the superiority of the new blade, which underlines the applicability of their design guidelines. Another paper by Hollands [9] investigates particular tip shapes of a research rotor using viscous CFD computations. He finds out that a slight dihedral is beneficial in forward

flight but not in hover. Baeder [10] solves the Euler equations for various rotor blade tips to find their effect on acoustics for simplified test cases. From an acoustic standpoint a forward sweep is better than a backward sweep for a high-speed impulsive noise. Further conceptual aero-acoustic investigations are made by Jones and Burley [11], who also look at various tip planforms with a CFD model and the employment of an acoustic code. They find that sweeping and tapering the blade at the tip as well as employing non-linear twist distributions reduce noise. Reason for this is the reduction in loading, which they also achieve by increasing the number of blades. Analyzing the effect of twist and rotor speed, Bousmann [12] identifies the need for operating each individual airfoil section at their optimal design point in order to globally raise the efficiency of a rotor. Schneider [13] looks at CFD results of various blade tip geometries for different hover test cases. He states that a high gradient of the twist distribution proofs to be beneficial as it improves the circulation of the blade. General thoughts on improving tilt rotors are made by Leishman and Rosen [14] emphasizing that a robust design is required. The rotor must handle various design points at once and making adjustments in RPM is essential, while the operational constraints are dynamic blade stall on the one hand and compression shocks on the other hand.

In opposition to these conceptual studies from the previous paragraph, numerical optimization is applied in many fields of helicopter design. An example is given by Bhadra and Ganguli [15], who reduce the hub loads and blade root loads of a helicopter rotor by changing the stiffness properties of the blade. Their analysis is based on a finite element method and a free-wake model along with a surrogate based optimization technique. Overall, they decrease hub loads by 16% to 22%. Another example is given by Glaz et al. [16], who reduce the vibratory loads of a helicopter in forward flight using various surrogate models and a genetic algorithm. They therefore modify structural properties of the blade to reach this goal. Another optimization is performed by Tritschler et al. [17], who mitigate the brown-out effect. They generate a locally accurate surrogate model based on polynomials and change the twist distribution of the rotor. Their finding is that a uniform inflow at a high loading is beneficial in their case. The underlying analysis tool is a free vortex model. Sajial et al. [18] decrease the hub vibratory loads of a helicopter by finding the control law for higher harmonic flapping. They compare the results found using a polynomial response surface, as well as an artificial neural network (ANN). They achieve a 27 % reduction in vibratory loads, and up to 45 % less power required for the rotor with flap control. Mani et al. [19] alleviate the effect of dynamic stall for a 2D airfoil problem for helicopters using an unsteady adjoint approach. Ahuja et al. [20] also optimize a 2D airfoil for dynamic stall and apply a time-spectral method with an artificial neural network coupled with proper orthogonal decomposition (POD) of the problem to speed up the design process. However, both papers demonstrate that single-objective optimization is not sufficient to create an overall well performing airfoil. The dynamic stall is relieved, but is paid for with a drag build up at regular flight conditions. Jones [21] demonstrates that a suitable parameterization is necessary as otherwise odd designs are obtained. He optimizes the aero-acoustics of rotor blade airfoils, but obtains impractical shapes. The airfoil surface is wavy and would be difficult to manufacture, which highlights the need to find a viable parameterization for the individual problem.

1.2.2 Aerodynamic Rotor Analysis

In order to understand such a complex system as a helicopter rotor, accurate and reliable methods for the analysis are required for its design. A good overview of most existing methods for computing the aerodynamics of helicopter rotors is given by Conlisk [22]. He describes the individual methods in detail and sketches the necessary assumptions and equations for each method. Here, only a brief introduction of available methods is given focusing on their strengths and weaknesses.

The first method for rotor analysis has been the momentum theory developed in the latter half of the 19th century by Rankine [23] and Froude [24] and is later on generalized by Glauert [25]. It is a rather thermodynamical consideration, which allows finding the required power for an ideal rotor at a certain thrust by only taking into account disc area and the density of the surrounding air. A more advanced approach is the blade element theory (BET), originally derived by Drzewiecki [26] for airplane propellers. The basic idea is to split the rotor blade into span-wise sections and compute the loads on them by integrating the 2D-forces of the according airfoil polars. There are many extensions of the original blade element theory as one key problem is to find the correct inflow through the rotor disc. For more details on the blade element theory, see Gessow [27] or Leishman [28].

As the design of a rotor blade itself is an aero-mechanical problem, it requires multiple disciplines to accurately model it. There is the aerodynamic part, which places the loads created by the motion of the blade through the air; the structure is elastically deformed under the aerodynamic and inertial forces; and there is the rigid-body motion, which covers the flight mechanics of the helicopter. The latter is required to account for blade dynamics as well as the trim settings obtained from keeping the helicopter in a desired flight path. This threefold task of aero-, structural and flight dynamics is accomplished with so called comprehensive codes. A general overview on the American first and second generation of comprehensive codes is given by Kunz [29], with popular programs such CAMRAD, UMARC, DYMORE and RCAS. The American comprehensive code CAMRAD II by Johnson [30] delivers good results for his analyzed flight tests of various rotor configurations. A European comprehensive code called HOST is validated by Arnaud et al. [31] for flight tests of a Puma AS-330. Benoit et al. [32] demonstrate the further advances of this code.

The BET, which is the most basic aerodynamic method found in comprehensive codes, is often coupled with the momentum theory in hover or linear inflow models in forward flight. A well-known example of these models is given by Pitt and Peters [33]. For these linearized models, a first order harmonic inflow distribution is assumed azimuthally throughout the rotor disc. Linear inflow models are highly efficient, but do not resolve specific details of the flow in particular in dependency of the underlying geometry. The next step towards more accurately resolving this is proposed by Kinner [34]. He suggests a modified potential theory to compute the pressure jump in the rotor plane assuming small perturbations with respect to the global inflow. Peters et al. [35] develop a finite state inflow model based on this approach. Basset et al. [36] incorporate changes to this theory accounting for in-ground-effects and implement it into HOST. This method is interpreted as an intermediate step towards wake modeling. It is computationally cheaper in comparison with other methods, but does not consider all the physical effects. In particular the tip vortex is not explicitly modeled, though the theory includes the bound circulation of the blades.

There are two major types of wake modeling to enhance the BET, either prescribed or free-wake models. A wake model describes the shed vortices of the blades through velocity potentials and computes the inflow velocities on the blade with the law of Biot-Savart. A prescribed wake follows a set path, based on semi-empirical knowledge, while a free wake model finds the equilibrium of the induced velocities and the wake geometry. Egolf and Landgrebe [37], [38] and Beddoes [39] derive prescribed wake models based upon experimental observations, which enables them to find the correct wake geometry based on semi-empirical parameters. Arnaud and Beaumier [40] show the advantage of a prescribed wake model over the simple Meijer-Drees inflow model in comparison with a flight test of a Puma AS-330 helicopter at slow and fast forward flight. Their wake model is based on the findings by Kocurek and Tangler [41]. Clark and Leipner [42] demonstrate that by employing a flexible free wake, the performance predictions are more accurate with respect to the momentum theory. Sadler [43] is also considered as one of the first movers to employ a free wake methodology to compute the wake geometry. Egolf [44] publishes a free-wake method that is based on a vortex-lattice approach rather than concentrating purely on the tip-vortex. Michea [45] incorporates a free wake model in the HOST code and finds a better accuracy for low speed flight in comparison with a prescribed wake model. Later on, Beaumier and Delrieux [46] predict aero-acoustic noise of the 7A and ERATO blade with Michea's free wake model. They also make a clear statement that the method has to be extended further for non-traditional planforms, such as the ERATO blade, due to the coarse representation of the blade by a single line. A comparison of different implementations is performed by van der Wall et al. [47], who validate their wake coupled comprehensive codes against wind-tunnel measurements of the HART II campaign. One result from this study is that prescribed wake models are competitive to free wake models, which is stated in the context of aero-acoustic and vibration investigations of a descent flight condition.

So far, the blade itself is represented by a line. This does not allow for the solution of three dimensional effects of the blade. An extension of free-wake codes is the incorporation of the blade surface by source/sink potentials. Ahmed and Vidjaja [48] validate the unsteady panel method (UPM) against wind-tunnel data of a BO-105 model rotor. Yin [49] utilizes this panel code for aero-acoustic predictions in descent flight conditions and he performs aerodynamic interaction studies with it as well. A similar method is employed by Massaro et al. [50] and Massaro and Andrea [51] for multi-objective optimization of rotor blades. In contrast to Yin, they have a viscosity correction in their code as otherwise the flow field is purely described by inviscid and incompressible potentials.

The current state-of-the-art method to predict rotor flows is CFD. Strawn at al. [52] prepare a good summary of the CFD activities in the field of rotorcraft. They mention how the first CFD computations are based on a transonic small perturbation (TSP) theory undertaken by Caradonna and Isom [53] for a hovering non-lifting rotor. As their method does not capture the rotor wake, later advances include a free-wake model to supply the vortex induced velocities into the CFD solution. An example of this inner-outer domain technique is given by Egolf and Sparks [54], who solve the full-potential equations instead of relying on the TSP. To obtain a better drag prediction, Beaumier et al. [55] employ a boundary layer code to include viscous effects in the simulation. Kroll [56] is the first to demonstrate that it is possible to compute the flow field of a hovering helicopter rotor without a wake-model solely solving the Euler equations in a rotating reference frame. Wake and Sankar [57] are the first to solve the Reynolds-

Averaged Navier-Stokes (RANS) equations for a helicopter rotor. The investigations so far are made in hover as the meshing requirements are limited. For forward flight, the Chimera technique helps to model multiple blades at once. In hover this is not necessary when only an isolated rotor computation is performed as periodic boundary conditions can be exploited. The Chimera technique embeds multiple grids into a background grid, therefore creating the possibility to model moving objects with non-moving objects in the same simulation. Among the first to apply this technique are Duque and Srinivasan [58] as well as Stangl and Wagner [59]. The inner-outer domain approach by solving the near-field with CFD and the farfield with a wake model is still utilized today [60] when simulating the HART II test case. Recent approaches are additionally fluid-structure coupled, such as the investigations by Pahlke and van der Wall [61]. They also stress the necessity to include viscous effects to correctly capture the pitching moment, which affects the blade torsion.

When solving the RANS equations for helicopter rotors, two major shortcomings are known. On the one hand, numerical dissipation arising by coarse meshes [62] or too dissipative numerical schemes cause the wake as well as tip-vortices to vanish too quickly in order to correctly capture the physics. On the other hand, modelling of stall phenomena is difficult with regular one and two equation turbulence models, especially for dynamic stall. A lot of research is ongoing concerning these two shortcomings. These approaches are not considered mature enough to compete with state-of-the-art RANS simulations, yet are briefly discussed for the completeness of this review.

Besides employing finer grids, the problem with the inherent numerical dissipation is tried to be overcome by employing higher-order schemes. These schemes generally introduce less numerical damping to the solution. Recent examples are Kowarsch et al. [63] or Potsdam et al. [64]. They demonstrate an improvement for their test cases; however, their computational effort is also tremendous. Another idea to better model the tip vortex is to transform the Navier-Stokes equations into the vortex transport equations by taking the curl on the momentum equations. By directly formulating the equations in terms of vorticity, vorticity is conserved better. Brown [65] solves the vortex transport equations to model hover and forward flight. However, they are formulated in an inviscid and incompressible manner, which does not represent all physics of the helicopter rotor. Thus, Whitehouse and Tadghighi [66] couple the vortex transport model with the RANS equations. They model the surroundings of the rotor with the vortex transport model, where the assumption of incompressibility and inviscid flow is mostly true. In the vicinity of the rotor blades the flow is computed with the Navier-Stokes equations. The coupling between both domains is then achieved through the Chimera technique. A third approach to refine vorticity prediction is called vortex confinement developed by Steinhoff and Raviprakash [67]. Here, an additional source term is added in the central region of an identified vortex region to avoid dissipating the vortex core. This method is also researched by Costes [68], who implements this method in a higher-order fashion to model the 7A rotor. A shortcoming of this method is that the results are very parameter dependent and have to be identified for each case individually.

The challenging topic of dynamic stall simulations is investigated by Smith et al. [69]. They investigate the effect of temporal and spatial adaptation on a pitching wing configuration. They find out that up to 70% time savings can be achieved in contrast to fixed mesh and time step sizes to obtain similar results. They advise to model turbulence with more advanced techniques such as Large-Eddy Simulations (LES). A com-

parison of different turbulence modeling methods is done by Choi et al. [70] for helicopter airfoils and three different flight conditions of the UH-60 Helicopter. They demonstrate that the predictions are improved through detached eddy simulations (DES). Kaufmann et al. [71] identify discrepancies for standard RANS approaches with a very fine resolution between numerical and experimental results of a dynamic stall case. They also stress the need for a high resolution to capture all effects.

As of now, the current state-of-the-art is the solution of the RANS equations for complete helicopter rotors in combination with two-equation turbulence models, such as the $k-\omega$-SST model by Menter [72]. They reliably help to compute the flows around various helicopter components in various scenarios, see [73], or complete helicopter configurations as demonstrated in [74] or [75]. Dietz and Dieterich [76] compare the results obtained by RANS equations along with a comprehensive code with a real flight test of a mass produced helicopter. Deviations of about 4.0% in required rotor power are stated with CFD being more conservative. Therefore, the application of RANS in aerodynamic design conditions proves to be sufficient, while for the simulation of aero-acoustics and off-design conditions more accurate methods are required.

1.2.3 Surrogate Techniques and Optimization Methods

A common approach to reduce the number of evaluations in a numerical optimization of any sort of system is the application of surrogate models. The idea of a surrogate model is to use only a small number of true function evaluations, here the analysis codes, and build a simplified, mathematical relation from these samples, which approximates the original function. The common concept is briefly described; first an initial design of experiments (DoE) is performed in order to obtain goal function/design vector pairs. Then the surrogate is created and a search within this model is performed for new sample locations, which are then recomputed with the simulation tool and are also added to the surrogate model. Different criteria for this infill points exist. They range from improving error estimates, goal function values as well as the expected improvement (EI) of the function. If the convergence criteria are met, the process is stopped. Otherwise the process starts over again at rebuilding the surrogate models with the newly computed samples.

The following paragraphs are covering the individual methods necessary for a surrogate based optimization framework required including variable-fidelity surrogate models.

Design of Expriments

Bhadra and Ganguli [15] present a selection of strategies for the initial sampling for surrogate based optimization problem of a helicopter rotor. Among these are the full factorial approach, Orthogonal Arrays and central composite designs. The full factorial approach is intuitive; in each spatial direction a uniform distribution of specified samples is placed. This leads to the "curse of dimensionality" as this sampling cube reaches the size of n^k samples with n the number samples per direction and k the number of design variables. An Orthogonal Array is simple cross of samples in a two dimensional problem and is similar to the data points required for a finite difference scheme including the center point. The central composite design is a blend of both. The Orthogonal Array is suitable when a set of polynomials has to be fit around a specific point, while

the full factorial will grant a more general view of the function. Monte-Carlo (MC) sampling strategies are based on random number generators for placing samples in the design space. One approach, especially for higher dimensional problems, is the Latin hypercube sampling (LHS). Similar to the full factorial, the design space is split into bins; only at randomly selected locations a point is placed. This allows the creation of a surrogate model with less samples than n^k points, but one problem that also occurs with MC sampling is that the distribution of points may cluster around a specific region. In order to avoid these phenomena, various metrics are formulated. Morris and Mitchel [77] propose a metric based upon the distance between the individual points and the number of occurrences of these distances. The lower this metric is, the better the sampling plan. Through permutation of points in the hypercube this metric may become better, an example of this process is given in Forrester et al. [78]. Another criterion for evaluating the quality of a hyper cube is the variance of the points, also referred to as energy; refer to Ju et al [79]. To enhance MC sampling plans, a technique called central Voronoi tessellation (CVT) exists, which associates bins with each individual point. Then, these points are de-clustered by injecting a second sampling plan of random points and sorting them into the bins for averaging the first set of points. Starting from a randomly uniform distribution, at some point an evenly distributed sampling plan is obtained. Lloyd [80] and MacQueen [81] develop such techniques, Ju et al [79] take these algorithms to blend these together and enhance them for parallel computing. A study by Romero et al. [82] demonstrates the superiority of CVT modified LHS compared to other sampling strategies. Only Quasi-Monte Carlo sampling strategies based on the prime-number sequences by Halton [83] and Hammersley [84] achieve similar results. Advantages of the Quasi-Monte Carlo sampling is that the numerical experiment is repeatable, but if the sampling plan misses certain aspects of the function, a repetition will not yield new results. Opposing this, the CVT based methods have their strength in generating different cubes of similar quality; however no (guaranteed) repetition is possible. Another downside of sequences is that they lack space fillingness in higher dimensional spaces.

Surrogate Modelling Methods

Often and commonly applied surrogate models in numerical optimization are polynomial response surface models, radial basis functions, Kriging and artificial neural networks. They are also combined with proper orthogonal decomposition, which allows interpolating many dependent variables at once.

Polynomial response surface models are the most basic approach to surrogate modeling. A polynomial function is set up as the ansatz for the true function and then the coefficients of the polynomial are determined by using a least squares approach. Collins [85] follows this approach in his work to cheaply model the vibration levels of a rotor blade and takes advantage of this in his optimization. Bhadra and Ganguli [15] investigate the best order for the polynomial, also to lower vibratory loads on rotor blades. Their conclusion is that second order polynomials in conjunction with an Orthogonal Array sampling or face-centered composite central designs yield the best results for their case.

The approximation through radial basis functions is based on the idea of correlating the sampled data with a new point that is to be predicted. They represent a linear combination of weights with correlation functions of each sample with the point to

be predicted. The weights themselves are determined by solving the linear system of equations that is set up by computing the correlation of each sample with another sample and set into relation to the response of this sample. One example of radial basis functions in SBO is given by Li et al. [86], who employ this method to optimize the packing profile of a molding process.

A more advanced method to radial basis functions approximation is Kriging. Originally created by Daniel G. Krige [87], the breakthrough of this method is achieved by Sacks et al. [88]. Kriging is interpreted as a mixture of a polynomial response surface and a radial basis function approximation. The purpose of the trend function, the polynomial response surface, is to roughly approximate the model. The difference or error between the prediction of the trend function and the true value of the sampled data points is then computed. This error is then re-predicted by a radial basis function approach depending on spatially dependent basis functions. Kriging is frequently applied in optimization as it features the possibility to compute statistical metrics such as the expected improvement of a goal function. The expected improvement becomes high in two regions of the surrogate model; either in regions where little is known and thus the chance of finding a new minimum is high, or in regions where the surrogate model predicts a potential minimum. Choosing the point with the highest expected improvement as the next infill point makes the surrogate based optimization more efficient than simply choosing the current best goal function value predicted by the surrogate model. This splits the task between exploring and exploiting the surrogate model. The exploring part ensures that a good global function approximation is achieved, while the exploitation is then the explicit search for the optimum. Jones and Welch [89] invent an Efficient Global Optimization (EGO) algorithm around Kriging and this global exploration technique.

One approach of parallelizing this process is made by Knowles [90], who extends the EGO algorithm to multi-objective optimizations. He proofs superiority compared to the NSGA-II from Deb et al. [91], an efficient genetic algorithm for multi-objective problems. Another effort of parallelizing EGO is made by Horrowitz et al. [92]. They compute the initial sampling in parallel, as well come up with a strategy to select multiple infill points at once. Their strategy selects local minima of the expected improvement function. To avoid a clustering of points, a safe guard measure is implemented to guarantee a user-specified distance from already computed points. Li [93] investigates various possibilities of combining sampling strategies and performing these in parallel to speed up airfoil optimizations. She decreases the overall turnaround time, but at the expense of an overall higher resource price. An important finding is also that it is difficult to find the appropriate sample settings prior to the optimization.

Kriging based optimizations are adopted by many researchers. Siller et al. [94] optimize a turbo machinery problem involving 231 parameters including multiple objectives. Forrester [95] demonstrates in his thesis the possibilities of Kriging and specific enhancements are discussed. He aims at reducing drag of a whole wing including the flap track faring and engine nacelles through the simulation with CFD. Jeong et al. [96] combine a genetic algorithm with a Kriging model to optimize airfoils using the EI based infill strategy.

In opposition to the deterministically based models, artificial neural networks try to mimic the behavior of a human brain. A neuron receives input signals from other neurons. Based on these signals, this neuron then sends a signal to other neurons. A common concept is to have an input layer of neurons, which features the input pa-

rameters of the problem, multiple layers of neurons behind this layer, and then a final output layer, which features the predicted value. A feed forward neural network will only transmit information from the previous layer to the next, while a feed backward network will propagate information in the reverse direction. The signal of each neuron is determined by the weighted sum of the input values, and the so called sigmoid function computes the output signal. The sigmoid function has almost binary behavior and switches between the function values zero and one when coming from negative infinity to positive infinity around the parameter value zero. Johnson [97] prefers artificial neural networks over Kriging for her optimization problems including 2D-airfoil drag reduction as well as the performance enhancements of a hovering and forward flying rotor blade. Her preference is based upon trial of these two methods.

Another technique in the field of surrogate modelling is the method of Proper Orthogonal Decomposition (POD). When the number of goal functions or system responses is greater than the number of samples given, POD reduces the dimensions to the number of samples to be interpolated. This enables a quick interpolation of large dataset, for example pressure distributions or complete flow fields. Most POD methods perform an eigenvalue analysis or singular value decomposition for the orthogonalization. A further speed up for the interpolation is achieved by dropping the smallest absolute singular values or eigenvalues from the dataset as they contribute the least to the prediction. This may lead to a regression, i.e. the data points are not exactly interpolated anymore. Robinson and Wilcox [98], as well as Robinson [99] map low- and high-fidelity codes through POD to account for the change of available parameters. Cai and Ladeinde [100] use POD for inverse design of an airfoil shape based on the so called Gappy POD approach. A compact version of the POD method is developed by Carlberg and Farhat [101] that stabilizes and speeds up the process for optimizing a structural problem.

In order to understand the differences, the advantages and disadvantages of each method, Peter and Marcelet [102] do a comparative study on artificial neural networks, radial basis function approximations and different types of Kriging. The demonstration test case is a turbo machinery optimization of a compressor blade. Their result is that Kriging methods are the most accurate surrogate models.

Variable-Fidelity Surrogate Models

Another aspect of the surrogate modeling field is given by the variable- or multi-fidelity methods (VFM or MFM), meaning that the models are built from two or more simulation or experimental sources. Fundamental ideas are explained in Gano's work [103], where he achieves over 60% resource savings using a variable-fidelity approach in an optimization. He investigates additive, multiplicative, and hybrid bridges, as well as scaling functions based on Taylor series and Kriging. Bridges are a type of modifier based on a surrogate itself, which models the difference, absolute or relative, between low and high-fidelity data. His engineering example is the design of a 2D high lift airfoil configuration. Co-Kriging is a variable-fidelity method, which joins low- and high-fidelity data in a large correlation matrix. Forrester et al. [78] combine a vortex code and a potential code by building a Co-Kriging model, which aids the design of a transonic aircraft wing. Their Co-Kriging method is based upon the difference of low- and high-fidelity points combined in a large correlation matrix. A different Co-Kriging approach is taken by Yamazaki and Mavripilis [104], who are di-

rectly composing the correlation matrix from low- and high-fidelity points through a weighting of them. This way, multiple fidelities are easily incorporated. They demonstrate the usefulness of this approach for multiple test cases, the most interesting being an aircraft configuration. Han et al. [105] compare various bridges including additive, multiplicative and hybrid bridges built from Kriging to come up with their own hybrid bridge methodology. Their version is the most complex one and also the most accurate choice. A different variable-fidelity method is proposed by Han and Görtz [106] referred to as Hierarchical Kriging. The idea of this approach is to first produce a low-fidelity Kriging model and employ this as the trend function for the high-fidelity Kriging model. This is also suitable for chaining multiple fidelities together as the low-fidelity model could be a Hierarchical Kriging model itself. With this approach they predict aero-loads for an experimental aircraft configuration. A similar Kriging method is proposed by Xiong et al. [107], who introduce an additional constant term to the trend model. They prove the applicability to synthetic test problems. They do not find the constant terms from the Kriging tuning process, but through simple linear regression of the high-fidelity data.

Applied Surrogate Based Optimmization

In the context of a surrogate based optimization, it is important to select a suitable optimizer or optimization strategy to locate the next infill points. Optimizers are either deterministic or heuristic. The deterministic optimizers are then subdivided into pattern search methods and gradient based approaches. The heuristic approaches are often evolutionary algorithms, which mimic a similar behavior to the biological process of reproduction. Roughly, most pattern search methods as well as gradient based approaches are good for local refinement of a goal function, while the evolutionary methods are suited to find the global optimum.

In the early beginning of surrogate based optimization, gradient based optimizers are combined with polynomial response surface models and a trust region management approach, see Alexandrov et al. [108]. The general mechanism of gradient based optimizers is that they compute a search direction based upon the gradient and then perform a line search on that path until no more improvement is found for the function. Then the gradient is re-evaluated. This process is repeated until the gradient becomes zero. A popular gradient algorithm is formulated by Fletcher and Reeves [109] belonging to the group of conjugate gradient methods. An example for trust region management is described in Eldred et al. [110]. The trust region management is based on a metric, which evaluates the accuracy of the surrogate model based on the prediction of the point and its actual value after it is evaluated. Depending on this metric, the maximum step size for the gradient based optimizer is limited, defining the trust region of the surrogate model. If the optimizer tries to evaluate a point outside this region, it is first sampled with the true function and fed back into the surrogate model. Gano et al. [111] and Robinson et al. [98] employed this technique. The trust region approach also works with local pattern searches such as the one by Hooke and Jeeves [112] or the simplex algorithm by Nelder and Mead [113]. While the first pattern search algorithm moves a multi-dimensional cross to its arm with the smallest goal function and shrinks it if the center of the hyper cross is the best point, the simplex algorithm has a simplex as its pattern. The simplex is transformed and moved towards the direction of the next best goal function. Overall, all these optimization techniques face the problem that they get stuck in a local minimum and due to their contraction mechanism will

stay in it.

More recent works use optimizers that claim to be more globally accurate and the surrogate model is built to capture the whole design space. The EGO algorithm by Jones and Welch [89] is driven by a branch and bound method, which simply divides the design space into smaller subsets of problems and estimates the lower limit of it. Then the subset with the smallest goal function value becomes the region to be split into more subsets and so continues the search process. Many examples in aerodynamic optimization choose genetic algorithms to search their surrogate models, such as [97], [50], [114], [115]. The general idea of genetic algorithms is to convert the design vector to a binary string. This string or chromosome is then mutated by exchanging zeros and ones arbitrarily in the string, crossing two individuals by cutting and pasting two strings together, or simply select the fittest among a subset of them. After a certain number of generations, the population contains the globally optimal configurations. A slightly different approach is taken by Storn and Price [116] performing vector based operations to cross individuals instead of the binary operations. They call their method differential evolutionary (DE) optimization.

1.2.4 Numerical Aerodynamic Optimization of Rotor Blades

Currently there are two major routes that are taken for the aerodynamic optimization of rotor blades. The first route to directly optimize the goal function, either by applying efficient gradient based method or employing simplified physical models. The other route is given by surrogate based optimization including CFD methods.

An example for the direct approach is presented by Lee and Kwon [117], who formulate the adjoint CFD solution of the hovering rotor problem to rapidly and directly find the least induced required power of a modified Carradonna Tung Rotor and a UH-60 blade. They parameterize the airfoil sections of the blade with Hicks and Henne bump functions. For the UH-60 blade, they improve the Figure of Merit (FM) by about 12.7%. This figure is based upon inviscid flow solutions without fluid-structural coupling.

A planform optimization with the adjoint approach is done by Dumont et al. [118] for a hovering rotor case with FM as the goal function. Their parameterization is based upon Bèzier Splines, and they claim a speed up of about seven, compared to finite differences, for a test problem of 25 design variables. One result is a gain of 6.8 points in FM, which is based on viscous, non-fluid-structure coupled computations.

Combining various strategies, Richez and Dumont [119] seek the best active twist setting. They use the adjoint formulation of the CFD solution and a gradient optimizer to find the optimum in hover, while in forward flight they employ a genetic algorithm in conjunction with the BET coupled with a wake model. Choi et al. [120] optimize a UH-60 rotor blade in forward flight using the adjoint method combined with a time-spectral approach. The sensitivities are computed in a de-coupled process.

An example of surrogate based optimization is given by Vu et al. [121], who optimize a Bo-105 blade with a Kriging based optimization technique. They modify the twist distribution, taper, as well as specific airfoil parameters. For their analysis, they combine the BET with a 2D airfoil flow solver. With eight design parameters, they increase the FM by 6.5%.

Massaro et al. [50] compute their samples with low- and mid-fidelity models based on a lifting-line model and a panel method coupled with a wake model for a multi-

objective problem with 20 design variables. The optima are then found in an artificial neural network by a genetic algorithm. They achieve a forward flight power reduction of about 12.0%. Massaro and Andrea [51] demonstrate the good correlation of the panel code with a classical state of the art CFD code for tilt rotor applications. A deviation is observed for higher thrust settings, where stall occurs.

Pioneering works in high-fidelity rotor optimization are performed by Le Pape [114] and Le Pape and Beaumier [122], who minimize torque for the 7A model rotor, as well as the ERATO blade and use a gradient based optimizer and alternatively a genetic optimizer to do so. A comparison between the BET and CFD is made for the tip sweep angle in a hover case, where it is found that the plain BET is too inaccurate for this task.

Imiela [123] investigates the effect of anhedral, chord, sweep, the transition point between the two airfoils, the starting position of the tip and twist for a modified 7A blade in hover and forward flight using CFD. His goal functions are FM in hover and the consumed power in forward flight of the rotor. He then combines these optimizations to find the Pareto frontier between these two goal functions through multiple weights of goals, only using twist as the independent variable. Imiela's optimization is set up with a modified EGO algorithm, which is also based on Kriging and finds the highest expected improvement of the goal function with the DIRECT algorithm. His main message is that fluid-structural coupling cannot be neglected for rotor optimization as otherwise the rotor planform comes out very differently. The blade sweep changes from forward to backward swept and a coupling of the twist parameter is recognized when including fluid-structural coupling effects in the simulation.

Johnson [97] takes the other surrogate option and applies an artificial neural network paired with a genetic algorithm to bring down the torque by modifying taper, twist, and anhedral in hover and forward flight for a UH-60 blade.

A total of 19 design variables is employed by Chae et al. [115] to the acoustics of a hovering helicopter. Their approach is also based on a Kriging surrogate model in conjunction with a genetic algorithm. Their resulting blades are swept, have a tapered tip and their thickness enlarged on the inboard stations of the blade. Leon et al. [124] focus on the enhancement of multi-objective strategies for rotor blade design. They propose a Nash game approach to efficiently solve the multi-objective problem. Practically, the Nash game splits the design variables among various sub-optimizations, which are carried out individually and are synchronized at a later point to include constraints. The sub-optimizations are performed similar to performing a multitude of weighted objective functions optimizations with different weights, similar to the route gone by Imiela [125]. They take advantage of the blade element theory enhanced with a prescribed wake model in forward flight and compute a viscous CFD solution in hover.

The first mover of variable-fidelity methods in the aerodynamic optimization of rotor blades is Collins [85] focusing on the required power as well as noise objectives. He exploits bridge functions and simple polynomial response surfaces in the optimization process. From his studies it is deduced that not necessarily all optima from the different fidelities coincide with each other. In a later work of Collins et al. [126] it is found that the combination of low- and mid-fidelity methods is beneficial. At the same time they call for a deeper investigation on this matter, as to what low-fidelity method is most suitable for what objective. Collins also states in [85] that good multi-objective optimization algorithms are necessary to find valid Pareto frontiers. Another work employing variable-fidelities in a multi-objective environment is done by Leusink et al. [127]. They do not combine the high and low-fidelity through a surrogate

model, but re-use the low-fidelity locations of Pareto optimal individuals to sample the high-fidelity design of experiments for the subsequent single-fidelity optimization in a smaller parameter space.

1.3 Thesis and Research Contributions

From the presented literature research, a hypothesis is derived:

Hypothesis: *The computational cost of the aerodynamic optimization of helicopter rotor blades is significantly decreased when models of varying-fidelity are employed in contrast to using purely high-fidelity results, while at least the same accuracy is maintained.*

The starting point for proving this hypothesis is given by the recent findings of Imiela [123] and Collins [126]. Imiela establishes a high-fidelity simulation framework based on fluid-structure coupled CFD computations while Collins first combines low- and mid-fidelity methods together with variable-fidelity surrogates for the optimization of helicopter rotors. In his work, Collins asks for a deeper investigation on the matter of what are suitable low-fidelity models as well as better variable-fidelity surrogate models and strategies.

Picking up these questions, this thesis contributes to the current research in the following points:

1. An optimization framework for the task of the aerodynamic planform optimization of helicopter rotors is set up. The most suited design of experiments, surrogate models and optimization strategies for this type of optimization are derived in Chapter 2 in the context of variable-fidelity methods. Additional novel ideas are derived in this work for sampling strategies for the high-fidelity simulation based upon the knowledge of the low-fidelity surrogate model, along with means to shrink the parameter space and to treat failing designs. A Pareto front based optimization technique is modified for the multi-objective surrogate based optimization.

2. A thorough investigation of available aerodynamic models of helicopter rotors is made and their fitness for rotor design is identified on a theoretical and experimental level with respect to the applicability in the variable-fidelity context. Therefore in Chapter 3 the requirements for such aerodynamic models are identified and how each model handles these different physical aspects. Lastly a parametric study along with a genetic optimization is performed with all listed models to determine their applicability for rotor blade optimizations.

3. A quantification of the gain of variable-fidelity over single-fidelity surrogate based optimization is made in Chapter 4 along with the proof of the hypothesis. This quantification has not been done before for helicopter rotor optimizations nor has the actual speed-up been proven. The level of the highest fidelity considered is also increased in contrast to Collins and Imiela. The wake is modeled within the background mesh instead of a free-wake method as opposed to Collins and in comparison with Imiela, no single blade approaches nor are coarsened meshes utilized. The forward flight is simulated with viscous CFD methods and all blades are embedded in a background mesh.

In Chapter 5, the findings and conclusions of this thesis are summarized.

2 Surrogate Based Optimization with Variable-Fidelity Methods

2.1 General Concept

Surrogate based optimization (SBO) is a sub-group of numerical optimization strategies and is an acceleration mechanism for regular optimization. Through mathematical abstraction of the true function, here the simulation code, a quickly to evaluate surrogate is created. The search of the optimum in this surrogate is a lot faster than in the true function. However, this mathematical abstraction is likely to lack in accuracy compared to the true function and thus has to be improved in regions of interest.

Three major components are required for successfully performing a surrogate based optimization:

1. **Design of Experiments**: This is an often underestimated task in which samples for constructing a surrogate model are selected. Unwise choices may lead to very ill-posed surrogates.

2. **Surrogate Models**: After initial data has been sampled, the proper surrogate model has to be selected. It has to fulfill the threefold task of being robust, efficient and accurate matching the individual problem at hand.

3. **Optimization Strategy**: While regular optimization directly searches the best goal function, in surrogate based optimization an adaptive sampling strategy has to be applied to find the best design. This also implies that the surrogate model has to be improved in regions of lesser interest, just to clarify that these regions are not of interest. Nevertheless, a proper optimization strategy is required when searching for such an infill point and a smart choice of optimization algorithms is required.

The flow chart of the process for the regular as well as the variable-fidelity surrogate based optimization process is sketched in Figure 2.1. The focus is laid first on the regular surrogate based optimization process, contained in the gray box. At the beginning of the process, the design of experiments is performed on the high-fidelity function from which the initial surrogate model is built. This surrogate model is then updated with new samples from the high-fidelity simulations which are chosen based on the infill criteria. Here, these are the expected improvement (EI) or the Pareto front. The simulation results of these selected points are then fed back to the surrogate model. This is done until a convergence criterion is met, such as reaching the design confidence. The extension to a variable-fidelity process is done by additionally creating the low-fidelity

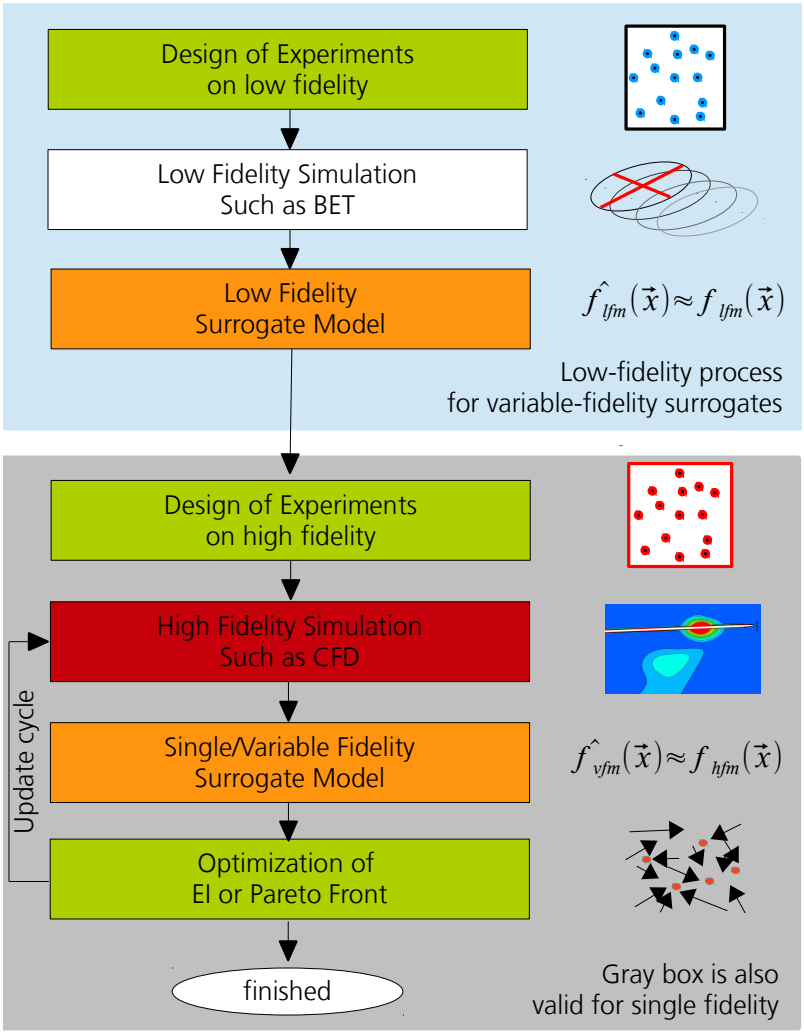


Figure 2.1: Surrogate Based Optimization (SBO), the regular process is contained in the gray box, while for the exploitation of the low-fidelity the whole process is required.

surrogate model using the same techniques as for the single-fidelity, see the blue box. A design of experiments is performed with low-fidelity simulations from which the low-fidelity surrogate model is abstracted. They are merged with the high-fidelity samples to generate variable-fidelity surrogate models replacing the single-fidelity surrogates during the high-fidelity update cycle.

This chapter gives an overview of the available and used methods, models, and algorithms for surrogate based optimization with the emphasis on Kriging surrogate models. A synthetic test problem is presented at the end for demonstration purposes.

2.2 Design of Experiments

The design of experiments (DoE) is the first initial task when setting up a surrogate based optimization. The purpose of the DoE is to distribute the samples $\mathbf{X}_s$ in such a way that a surrogate with a good first approximation of the actual function can be created. Through an iterative update process of the surrogate model, the regions of optimal designs are resolved later on. A good DoE will reduce the number of sam-

ples required in the adaptive sampling process. This is often performed in a sequential manner. For the DoE the evaluation of the samples can be easily performed simultaneously if sufficient parallel computing power is available. Therefore, the turn-around times can be further minimized.

2.2.1 Full Factorial Design

The full factorial design is interpreted as the brute force mechanism among the sampling plans, but it is easy to understand and set up. In each direction a set number of points is scanned. This is done for all dimensions ending up in a recursion over the dimensions. This technique becomes very expensive for greater dimensions. For example, the use of five points per direction for a problem depending on four variables, will require $5^4 = 625$ points to evaluate, while five points for eight dimensions will lead to $5^8 = 625 \cdot 625 = 390,625$. This exponential growth is referred to as the curse of dimensionality.

The advantages of this sampling strategy are that it is uniformly distributed and being deterministic. The downsides are that the projection of the samples onto a plane results in overlaying points. Periodic problems could just be sampled at re-occurring system responses. [128]

2.2.2 Latin Hypercube Sampling (LHS)

A Latin hypercube decreases the number of required samples for most problems, while still capturing most information required for building the surrogate model. The idea behind a Latin hypercube is to divide the design space into bins, and randomly distribute the points in such a way that when projected onto any 2D plane, only one point is contained in each bin. The downside of a regular Latin hypercube is that points can accumulate in a certain region, while other regions remain unsampled.

A simple strategy mentioned in [77] and [128] to improve a Latin hypercube is to randomly permute the numbers in each design vector of the cube and then re-evaluate the cube with the Morris & Mitchell criterion. If the cube is better, store it as the best cube and continue with the permutation in any case. By randomly choosing the vector to be permuted, as well as the columns, entropy in terms of a chaotic distribution is created. This is repeated until a convergence criterion is met or a total number of iterations is performed. This strategy yields an optimized Latin hypercube (OLHS).

2.2.3 Quasi Monte-Carlo (QMC) Sequences

A Monte-Carlo sequence is based on random events, while the Quasi Monte-Carlo sequences try to mimic this behavior. Two popular Quasi Monte-Carlo sequences by Halton [83] and Hammersley [84] are based on prime numbers. Therefore, the location of samples is deterministic. The advantage of such sampling plans is the possibility to repeat the DoE and obtain the same results again. Yet, this may also be an unwanted effect as samples can be placed in unsuitable regions of the problem.

The strength of the Halton sequence is that when the number of samples is increased, the previous cube is a subset of the new cube. The other way around is beneficial in the variable-fidelity context, the highly resolved cube for the low-fidelity method can

be reduced for the high-fidelity data and thus there are points where low- and high-fidelity data is computed. However, in terms of distributing points in a more uniform way, the Hammersley sequence is slightly better than the Halton sequence as found out by Romero et al. [82]. A weakness in higher-dimensional spaces is that the prime number based sequences lead to an inadequate distribution of points as Robinson and Atcitty [129] demonstrate for the Halton sequence.

2.2.4 Central Voronoi Tessellation (CVT)

The purpose of tessellation is to create an even clustering of items. An example can be the honeycombs of bees, where the same pattern is repeated over and over again. For the creation of spacing filling hypercubes, Ju et. al [79] present multiple clustering techniques to achieve space filling DoEs. One of their algorithms is a blend of MacQueen's Method [81] and Lloyd's Method [80]. It basically starts of an initial random sampling and then places a new random cube on top of that. The points of the second cube are then clustered and the points from the initial random sampling are then updated with the average vector of its respective cluster and its own vector.

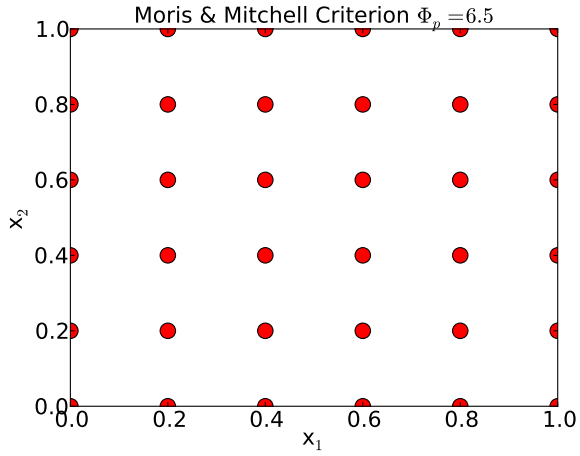
For the here presented problems, it is observed that a good initial cube is beneficial before the clustering is started. Thus, the Hammersley sequence is chosen in spaces with less than five dimensions, while an optimized Latin hypercube is applied for higher dimensional spaces. The overlay cube is a regular Latin hypercube, which has eight times more samples than the initial cube. The clustering process is repeated a 100 times or stopped if no improvement is found over the last ten cubes. The best cube is stored during each cycle and later on the points are evaluated with the truth function.

2.2.5 Quality Metric: Morris & Mitchell Criterion (MMC)

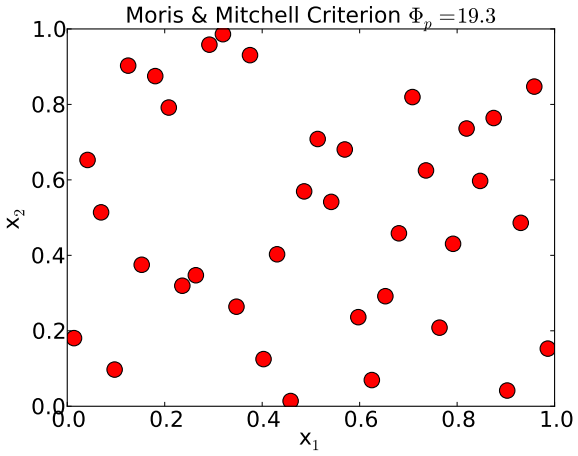
Morris and Mitchell [77] develop a criterion which tries to quantify the quality of hypercubes in order to compare different hypercubes and decide upon the best one. The goal of a good DoE is to be space filling. This means that all points should be as far away as possible from each other. Also, when looking at the values of the points in one direction, their projections should not overlap. The latter guaranties that periodic problems are sampled well enough by placing samples at unequal distances. The MMC criterion, which tries to quantify the space fillingness of a hyper cube $\mathbf{X}$ is given by:

$$\Phi_p(\mathbf{X}) = (\sum_{j}^{m} J_j d_j^{-q})^{\frac{1}{q}} \tag{2.1}$$

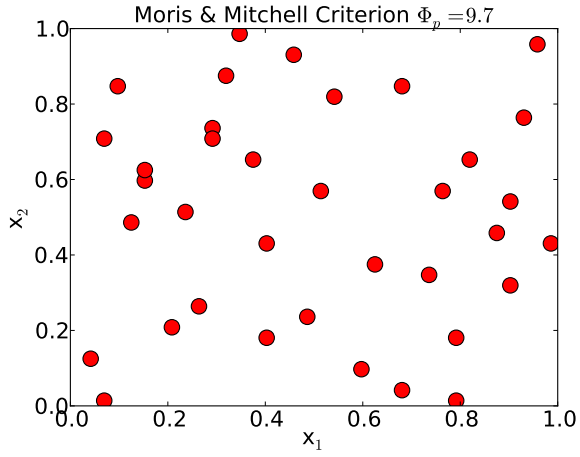
with J being the number of occurrences of the unique distances d in a hypercube and q an exponential weight. The distances may be computed using the one norm in order to reduce the computational effort [128]. The best DoE is given for a minimum number of occurrences J and maximum distances d. Thus a small Φ_p is favorable, while q either shifts the emphasis onto the occurrences (small q) or the distances (large q). Here, q is chosen to be 20 after initial trial and error.



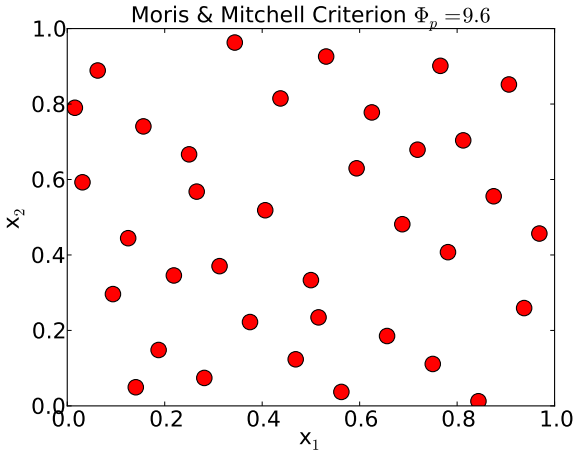
(a) Full Factorial Design



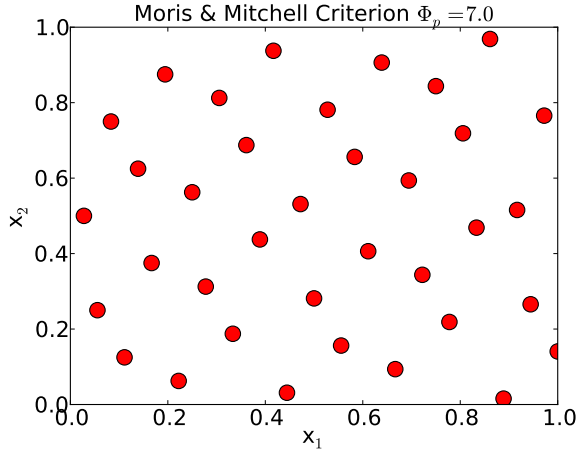
(b) Latin Hypercube



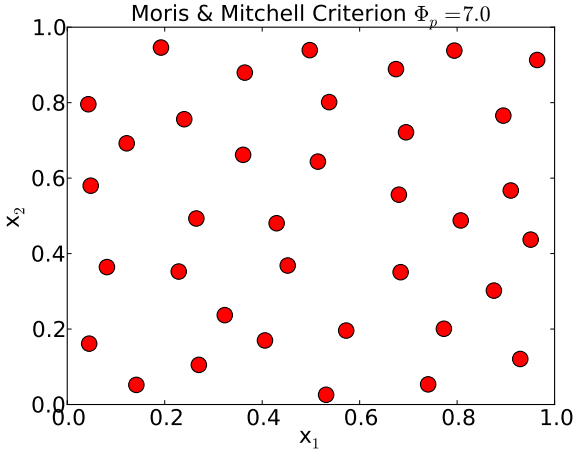
(c) Optimized Latin Hypercube



(d) Halton Sequence



(e) Hammersley Sequence



(f) Central Voronoi Tesselated-Cube

Figure 2.2: Different sampling strategies for a unit square and 36 sampling points.

2.2.6 Comparison of Different Techniques

For a better understanding, a quick comparison of the various design of experiments is presented in Figure 2.2 depicted in a similar graphical fashion as done by Romero et al. [82]. Here, 36 samples are generated in a two dimensional space. The quality according to the Morris & Mitchel criterion (MMC) is well reflected by the graphical representation. The full factorial cube is the best deterministic sampling technique for this problem as it has the lowest MMC, the Hammersley sequence following up and CVT almost equally good. For two dimensional non-periodic problems, the full factorial features a viable sampling strategy, but beyond that it is not recommended. The Hammersley sequence is a good alternative up to four dimensions. For larger design spaces, the CVT cubes prevail. In two dimensions, this approach has similar quality, deduced from Figure 2.2, yet this varies depending on the randomization. Despite this fact, the CVT cube is the preferred choice in the work, in particular as it starts from a Hammersley sequence in lower dimensional spaces. For simplicity, it will be referred to as a random sampling technique later on.

2.3 Surrogate Modeling

The general idea of surrogate modeling is to approximate the results of a complex function in order to reduce the need to evaluate this complex function each time a new set of independent variables is investigated. This complex function may either be a physical experiment or a numerical simulation. The basic formulation of a surrogate model is:

$$y = f(\vec{x}) \approx \hat{y}(\vec{x}) \tag{2.2}$$

where y is a function depending on the input vector $\vec{x}$, the surrogate function $\hat{y}$, which is to be determined also dependent on $\vec{x}$. In this work, the focus is laid upon the Kriging surrogate model family, as in contrast to other surrogates models, they yield valuable statistical metrics, see Section 1.2.3. After introducing Kriging, the statistical surrogate based optimization metrics are discussed to finish this section with variable-fidelity methods.

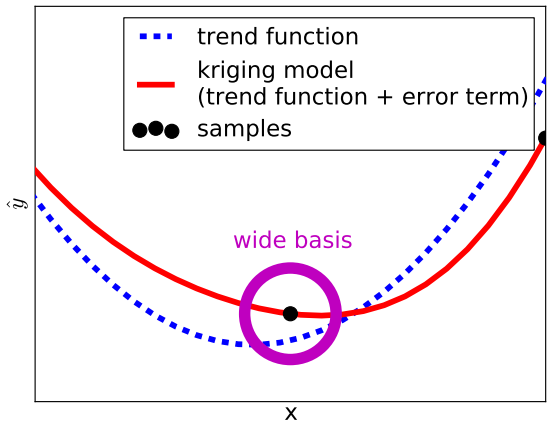
2.3.1 Kriging

Kriging is developed by Krige [87] in 1951. It is interpreted as a combination of a trend function, e.g. a polynomial regression model, and a radial basis function (RBF) approximation of the error between trend function and sample points:

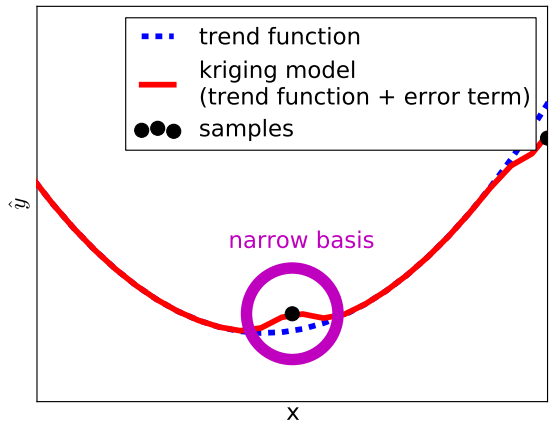
$$\hat{y}(\vec{x}) = \hat{f}_{trend}(\vec{x}) + \hat{\epsilon}_{rbf}(\vec{x}) \tag{2.3}$$

where the first term, $\hat{f}_{trend}$, is a polynomial regression model and the second part, $\hat{\epsilon}_{rbf}$, a radial basis function approximation. For details on polynomial regression models and radial basis function approximations, see Section A.1.1. Equation (2.3) is expanded to:

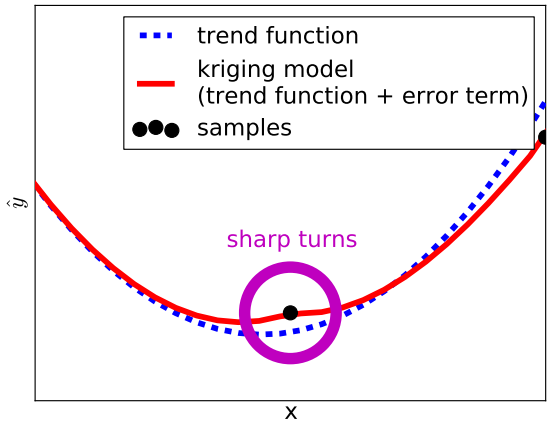
$$\hat{y}(\vec{x}) = \vec{f}(\vec{x}) \cdot \vec{\beta} + \vec{\psi}(\vec{x}) \cdot \vec{W} \tag{2.4}$$



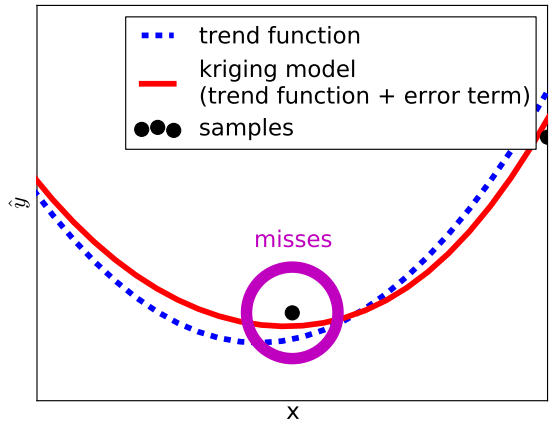
(a) Kriging with $\theta = 10, p = 2, \Lambda = 0$



(b) Kriging with $\underline{\Theta = 1000}, p = 2, \Lambda = 0$



(c) Kriging with $\theta = 10, \underline{p = 1}, \Lambda = 0$



(d) Kriging with $\theta = 10, p = 2, \underline{\Lambda = 0.1}$

Figure 2.3: Different settings for regular Kriging. A second order trend function is utilized along with four sample points, only two are shown. The chosen values of θ, p and Λ may not be representative for interpolation purposes.

with $\vec{f}$ the regression vector and $\vec{\beta}$ the corresponding coefficients, ψ the correlation vector and $\vec{W}$ the vector of weights. For a second order trend function, the regression vector $\vec{f}$ is a list of polynomial functions. For a one dimensional problem the vector for a second order regression is:

$$\vec{f} = (1, x, x^2)^T \tag{2.5}$$

The regression coefficients for the Kriging model are determined through a least squares problem generalized with the correlation matrix $\mathbf{\Psi}$ arising from the derivation of Kriging:

$$\vec{\beta} = (\mathbf{F}^T \mathbf{\Psi}^{-1} \mathbf{F})^{-1} \mathbf{F}^T \mathbf{\Psi}^{-1} \vec{Y}_s \tag{2.6}$$

The correlation vector $\vec{\psi}(\vec{x})$, as well as the correlation matrix $\mathbf{\Psi}(\mathbf{X}_s)$ are built using the following radial basis function:

$$\psi(\vec{x}_i, \vec{x}_j)_{Kriging} = \exp(-\sum_k \theta_k |x_{i,k} - x_{j,k}|^{p_k}) \tag{2.7}$$

which differs from the regular RBF approximation by the two tuning parameters θ and p for each spatial direction k. i and j are the indices for the rows and columns of $\mathbf{\Psi}$. The weights vector $\vec{W}$ is also constructed differently in comparison to regular RBF approximations. Instead of purely taking the sampled responses $\vec{Y}_s$ into account, the deviation or error from the regression model is included:

$$\vec{W} = \mathbf{\Psi}^{-1} \cdot (\vec{Y}_s - \mathbf{F}(\mathbf{X}_s) \cdot \vec{\beta}) \tag{2.8}$$

With $\mathbf{F}$ the regression matrix of the sample points. It is noted that it is not necessary to directly invert the correlation matrix $\mathbf{\Psi}$, but solving the linear system of equations through a Cholesky decomposition features a numerically more efficient and accurate alternative.

For the classical Kriging correlation function, often only θ_k is tuned and p is set to 2, which is also done here. Yet, other RBFs can be implemented for Kriging, such as a cubic plate spline as done by Han et al. [105]. The noise coefficient Λ may be added to the diagonal of the correlation matrix $\mathbf{\Psi}$ for stability and regression of the model. Suitable values for θ, p as well as Λ are found by maximizing the likelihood function L as written in [128]. The approximated concentrated likelihood function $\ln(L)$ is utilized as it decreases the computational effort:

$$\ln(L) \approx -\frac{n}{2} \ln(\hat{\sigma}^2) - \frac{1}{2} \ln |\mathbf{\Psi}| \tag{2.9}$$

with the model variance $\hat{\sigma}^2$ defined by:

$$\hat{\sigma}^2 = \frac{(\vec{Y}_s - \mathbf{F}\vec{\beta}) \cdot \mathbf{\Psi}^{-1} \cdot (\vec{Y}_s - \mathbf{F}\vec{\beta})}{n} = \frac{(\vec{Y}_s - \mathbf{F}\vec{\beta}) \cdot \vec{W}}{n} \tag{2.10}$$

For a better understanding of the Kriging process, Figure 2.3(a) demonstrates how the trend function is corrected by the error terms modeled by the RBF approximation. If θ is enlarged, a plot such as Figure 2.3(b) is obtained. The correlation function becomes steeper and thus the error correction becomes more locally. In opposition to this, the effect of a single basis becomes more global, if θ becomes small. A reciprocal behavior is seen for p in Figure 2.3(c). Decreasing it makes the correlation function also steeper while at the same time it becomes unsmooth. The last picture, Figure 2.3(d), demonstrates the effect of the noise constant Λ. This model is regressive and the Kriging approximator does not go directly through the initial sampling points anymore. Toal et al. [130] emphasize the necessity of properly tuning the Kriging model within a surrogate based optimization environment, as otherwise the accuracy may suffer and the number of samples required to correct this grows unproportionally fast. They employ a genetic algorithm and pose the tuning process as a numerical optimization problem. Here, the optimization strategies explained in Section 2.4.1 also tune the hyper parameters of the Kriging model. The parameter ranges applied in this thesis are always

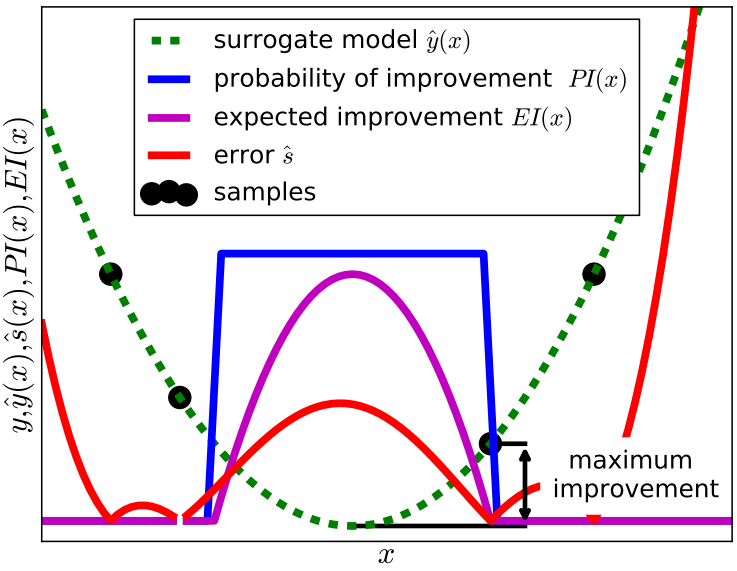


Figure 2.4: Sketch of the different surrogate model metrics

listed with their respective problems. Generally, higher θ values are often applied in lower dimensions, while the opposite is true for higher dimensional design spaces.

2.3.2 Exploration and Exploitation

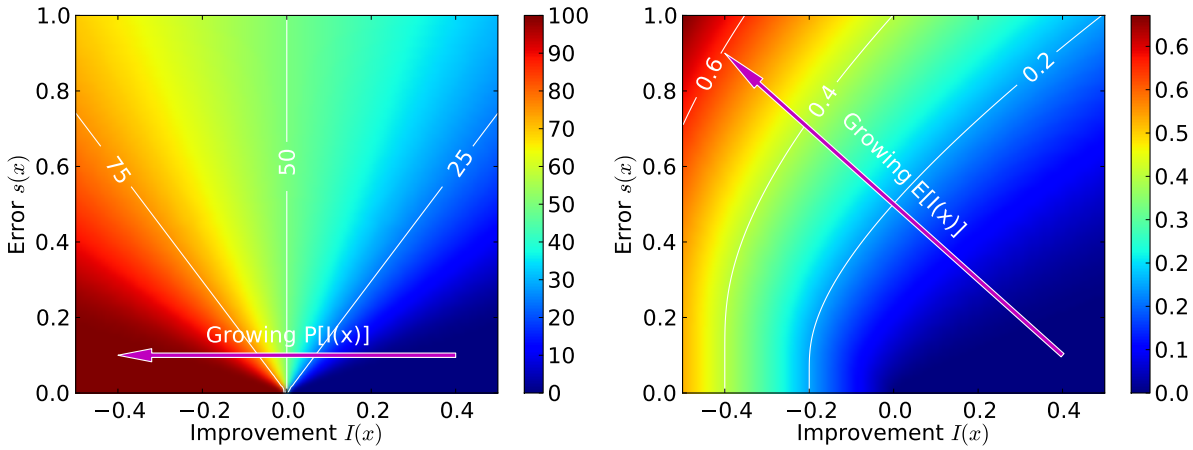
Kriging allows the quantification of uncertainties of the surrogate model. This helps to improve the surrogate model in regions where these uncertainties are large by placing new samples there, effectively reducing these uncertainties. This leads to techniques that help to either increase the accuracy of the surrogate model or optimize the approximated function. Balancing the creation of a globally accurate surrogate model with locally refined minima is often referred to as 'exploration versus exploitation' in the literature. The mean squared error (MSE) of a statistical model like Kriging is derived by Sacks et al. [88] and is estimated for a point in the parameter space by:

$$\hat{s}^2(\vec{x}) = \sigma^2[1 + \Lambda - \vec{\psi}^T\mathbf{\Psi}^{-1}\vec{\psi} + (\vec{f} - \vec{\psi}^T\mathbf{\Psi}^{-1}\mathbf{F})(\mathbf{F}^T\mathbf{\Psi}^{-1}\mathbf{F})^{-1}(\vec{f} - \vec{\psi}^T\mathbf{\Psi}^{-1}\mathbf{F})] \quad (2.11)$$

The error $\hat{s}$ is proportional to less sampled areas. The term $\vec{\psi}^T\mathbf{\Psi}^{-1}\vec{\psi}$ will be small for points far away from the sampling data set as the correlation with them will be small, while $\vec{f} - \vec{\psi}^T\mathbf{\Psi}^{-1}\mathbf{F}$ becomes large if the trend at the sampled points is far off from the sampled responses. Computing additional samples in these regions will drive down the MSE.

Another uncertainty metric is the Probability of Improvement (PI), which is more of an intermediate value for the expected improvement:

$$P[I(\vec{x})] = \frac{1}{\hat{s}\sqrt{2\pi}} \int_{-\infty}^{0} \exp(-I(\vec{x})^2/(2s^2))dI \quad (2.12)$$



(a) PI (b) EI

Figure 2.5: Probability of Improvement (PI) and Expected Improvement (EI)

with the improvement I defined as the difference between goal function value y_{min} of the best sampled and a predicted point $\hat{y}(\vec{x})$:

$$I(\vec{x}) = \Delta y = y_{min} - \hat{y}(\vec{x}) = \min(\vec{Y_s}) - \hat{y}(\vec{x}) \tag{2.13}$$

It may be rewritten with the numerical error function erf as:

$$P[I(\vec{x})] \quad = \frac{1}{2}\left[1 + erf(\frac{\Delta y}{\hat{s}\sqrt{2}})\right] \tag{2.14}$$

The probability of improvement quantifies the chances of $\hat{y}$ at the location $\vec{x}$ being better than y_{min}, the currently best sampled response. A key metric in surrogate based optimization is the expected improvement (EI) defined as:

$$E[I(\vec{x})] \quad = P[I(\vec{x})]\Delta y + \hat{s}\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(\Delta y)^2}{2\hat{s}^2}\right] \tag{2.15}$$

The expected improvement meassures the potential reduction of the goal function value under uncertainty and thus balances model error with goal function improvement. Jones et al. [89] build an optimization framework based on Kriging and its ability to compute the EI naming the method Efficient Global Optimizer (EGO), while Forrester et al. discuss this method at length [128].

For a greater clarity, Figure 2.4 sketches the different functions of $\hat{y}(x)$, $\hat{s}(x)$, $PI(x)$, and $EI(x)$ for a simple parabolic function in depency of the given samples points. In a different representation, Figure 2.5(a) shows how the probability of improvement behaves for a given improvement and error estimate, while Figure 2.5(b) depicts the behavior for the expected improvement. For a large (estimated) improvement, meaning a large negative value of $I(\vec{x})$, and a finite error $\hat{s}$, the probability of improvement grows towards a 100%, indicated by the red area on the plot. If the error $\hat{s}$ grows towards infinity for a finite improvement $I(\vec{x})$, the probability idles at 50%. Lastly, if no improvement is found anymore in the surrogate model, the probability becomes smaller

the more accurate the surrogate model becomes. Basically, an already sampled point has an improvement of zero with an error of zero, thus the probability of improvement is also 0%, compare Figure 2.4. The same is then true for the expected improvement. The expected improvement is almost equal to the improvement itself, if the error is small, which is also found in Figure 2.5(b) for the vertical lines on the bottom left corner of the plot. If the error becomes large, the expected improvement becomes larger for a constant improvement, as the error is added to the likely improvement. This guarantees that regions of high error are also visited during optimization.

2.3.3 Constraints in Surrogate Based Optimization

A common approach to handle constraints in non-gradient based optimizations is to apply a penalty value to the goal function in case constraints are not met. As in this work the EI is used for spotting potential minima, a different approach becomes available as presented in Forrester et al. [128]. Let the constraint be $g(\vec{x}) < g_{max}$, where $g(\vec{x})$ is the value of the constraint function, and g_{max} the value that should not be exceeded. Then by constructing a Kriging based surrogate of $g(\vec{x}) \approx \hat{g}(\vec{x})$, the probability of $g(\vec{x}) < g_{max}$ is computed in a similar fashion as $P[I(\vec{x})]$. Instead of using y_{min} and $\hat{y}(\vec{x})$, $g(\vec{x})$ and g_{max} are inserted into Equation (2.14) with the error $\hat{s}_g$ from the surrogate of the constraint function. The constraint EI is then a product of the unconstrained EI and the probability of not violating the constraint.

$$E[I(\vec{x}) \cap g(\vec{x}) < g_{max}] = E[I(\vec{x})]P[g(\vec{x}) < g_{max}] \tag{2.16}$$

Thus, the original EI is simply multiplied by the probability that the constraint is less than its maximum allowable value. To inverse the constraint, e.g. to enforce a minimum allowable value, the constraint is negated, similar to inverting an optimization problem. Equality constraints are handled by allowing a specific error ϵ_g, thus the problem is reformulated as $|g(\vec{x}) - g_{max}| < \epsilon_g$, allowing to further pose the problem as an inequality constraint. For multiple constraints, the individual probabilities are all multiplied together with the EI function.

During the multi-objective optimizations, the statistical metric of EI cannot be applied anymore. Thus, the points where the surrogated constraint is explicitly violated are marked as invalid and are not considered any further by the optimization process.

2.3.4 Design and Pareto Confidence

One problem with surrogate based optimization is how to decide that the optimization is finished. There are multiple stopping criteria; either the accuracy of the surrogate, a sufficiently small change in the goal function or the design vector variables over consecutive update steps, or a specified number of iterations, where no improvement is found. Except for the first criterion, none of them guarantees that the global optimum within the model is found. Validating that the surrogate is globally accurate is usually prohibitively expensive, and thus Xiong et al. [107] propose the design confidence as a stopping criterion. The design confidence quantifies the probability that a selected candidate solution $\vec{x}^*$ will not be better than the optimal solution in the surrogate $\vec{x}_{min}$ by a specific threshold value H. The solution $\vec{x}^*$ is chosen to be the best truth sample

so far. In practical terms, the value of $\hat{y}^*$ and $\hat{y}_{min}$ are computed using the surrogate model and together with their Kriging errors $\hat{s}^*$ and $\hat{s}_{min}$, the Design Confidence (DC) is computed by:

$$DC[(\vec{x})] = P[y^*(\vec{x}) - \hat{y}_{min} < H] = \frac{1}{2}\left[1 + erf\left(\frac{H - (\hat{y}^*(\vec{x}) - \hat{y}_{min})}{\sqrt{2(\hat{s}^{*2}(\vec{x}) + \hat{s}^2_{min} - \hat{s}_{min}\hat{s}^*)}}\right)\right] \tag{2.17}$$

Xiong et al. propose that a user defined DC should be reached in two consecutive steps to be certain that no more improvement, within the tolerance H, will occur. The expensive part of obtaining the design confidence is that a sub-optimization is required, where the current minimum $\hat{y}_{min}$ of the surrogate is searched. While this is almost the same as optimizing EI, it basically doubles the effort in the surrogate model. If constraints are set in the optimization, the probability of a non-violation needs to be multiplied with the DC.

As the design confidence is only valid for single-objective problems, in this work a metric for a found Pareto front is derived from the DC, referred to as Pareto Confidence (PC). The Pareto front is searched within the current surrogate models. The set of points representing the Pareto front are then checked against a **static** improvement defined by the user, here $I = 0.01 = H$. Similar to the DC, the probability that this point does not improve by this threshold is checked. For the PC, the probability is computed for each goal function and the smallest probability of all the goal functions and points of the considered Pareto set then features the Pareto confidence. With this metric, it should be ensured that no point improves more than 0.01 in any goal function. As an equation, the Pareto confidence is written as:

$$PC(\mathbf{X}_{PF}) = min_l(min_i(P_l[I_l(X_{PF,i}) < H])) \tag{2.18}$$

with X_{PF} the set of Pareto optimal points found in the surrogate model, i their index counter, and l the index of the goal functions. min takes the minimum of each set respectively. This ensures that the whole Pareto set fulfills this condition in both goal functions.

A side note, both the DC and the PC may yield false results, if the error prediction is off. This happens for example if Kriging is made regressive by adding the noise constant Λ onto the diagonal of the correlation matrix $\mathbf{\Psi}$.

2.3.5 Variable-Fidelity Modeling

The goal behind variable-fidelity modeling is to create an accurate surrogate model for high-fidelity data with fewer resources than required for the single-fidelity surrogate model by generating most of the predictor with a low-fidelity model and only calibrating the predictor with a few high-fidelity samples. The general assumption is that the low-fidelity simulation is faster than the high-fidelity simulation with often sacrificing the accuracy. However, it should closely match with the high-fidelity results in order to be most efficient for the variable-fidelity surrogate based optimization. This statement also leads to the variable fidelity paradox postulated by the author:

"It is desirable that the low-fidelity resembles the high-fidelity model at reduced cost of the high-fidelity model as close as possible in order to benefit the most from the variable-fidelity approach. However, a physical model is not a low-fidelity model anymore, if it exactly reproduces the results of high-fidelity model (or reality). If this is the case, the variable-fidelity approach becomes obsolete as the original high-fidelity model is not necessary anymore."

For Kriging surrogate models, there are three major options for variable-fidelity modeling, Hybrid Bridges, Co-Kriging and Hierarchical Kriging, which are discussed in the following.

Hybrid Bridges

The logic behind bridges is that from a low-fidelity model a specific mapping can be found to map the low-fidelity results to the high-fidelity results and allow making high-fidelity predictions, where no high-fidelity samples exist. The work of Gano [103] lists different options for bridges. There are two major types of bridges, additive and multiplicative bridges. Additive bridges offset the low-fidelity:

$$\hat{y}(\vec{x}) = \hat{y}_{lfm}(\vec{x}) + \gamma(\vec{x}) \tag{2.19}$$

while multiplicative bridges are flexible factors to the low-fidelity:

$$\hat{y}(\vec{x}) = \rho(\vec{x})\hat{y}_{lfm}(\vec{x}) \tag{2.20}$$

In both cases, the bridges $\gamma(\vec{x})$, and $\rho(\vec{x})$ may be surrogate models themselves, where Gano applies regression models. It is then tried to minimize the differences between high- and low-fidelity data by adapting the bridges. Gano proposes a Hybrid Bridge being mix between the other two bridges through the application of a weighting constant. Opposing to this, Han et al. [105] come up with a Hybrid Bridge version based on a Kriging surrogate:

$$\hat{y}(\vec{x}) = \rho(\vec{x})\hat{y}_{lfm}(\vec{x}) + \gamma(\vec{x}) \tag{2.21}$$

where $\rho(\vec{x})$ is a polynomial regression model, and $\gamma(\vec{x})$ is a Kriging model. The advantage of Han's approach over Gano's is that the high-fidelity data is exactly interpolated due to the Kriging model and is thus the here investigated Hybrid Bridge model.

Co-Kriging

Co-Kriging is utilized by Forrester et al. [78], Yamazaki and Mavripilis [104] as well as by Han et al. [131]. In Co-Kriging the model fusion is done by correlating high and low-fidelity data in a greater correlation matrix. The approach by Forrester et al. [128] is explained briefly here, which differs slightly from the implementations of Yamazaki and Mavripilis [104] as well as the one of Han et al. [131]. Still, the underlying concept remains the same.

At first, an ordinary Kriging model $\hat{y}_{lfm}$ is fitted to the low-fidelity data. With this model a difference vector $\vec{d} = \vec{Y}_{s,hf} - \rho\vec{\hat{Y}}_{lfm}$ is computed at the high-fidelity data locations. If no low-fidelity data exactly exists at the locations of the high-fidelity data, then these are interpolated at the location $\mathbf{X}_{s,hf}$. This vector $\vec{d}$ is then also fitted by an ordinary Kriging model, but during this likelihood optimization, ρ is also determined. With both models created, the joint covariance matrix $\mathbf{C}$ is constructed as follows:

$$\mathbf{C} = \begin{pmatrix} \hat{\sigma}^2_{lfm}\mathbf{\Psi}_{lfm}(\mathbf{X}_{lf}, \mathbf{X}_{lf}) & \rho\hat{\sigma}^2_{lfm}\mathbf{\Psi}_{lfm}(\mathbf{X}_{lf}, \mathbf{X}_{hf}) \\ \rho\hat{\sigma}^2_{lfm}\mathbf{\Psi}_{lfm}(\mathbf{X}_{hf}, \mathbf{X}_{lf}) & \rho^2\hat{\sigma}^2_{lfm}\mathbf{\Psi}_{lfm}(\mathbf{X}_{hf}, \mathbf{X}_{hf}) + \hat{\sigma}^2_{d}\mathbf{\Psi}_{d}(\mathbf{X}_{hf}, \mathbf{X}_{hf}) \end{pmatrix} \tag{2.22}$$

with $\hat{\sigma}^2$ the model variance of either the low-fidelity model (lfm) or the difference model (d). The individual correlation matrices $\mathbf{\Psi}$ are computed using the respective hyper parameters θ and p. Note, the relation between covariance matrix $\mathbf{C}$ and correlation matrix $\mathbf{\Psi}$ as in regular Kriging (Equation (2.4)) is $\mathbf{C} = \sigma^2\mathbf{\Psi}$ for single fidelity Kriging. To make a prediction, a covariance vector is created:

$$\vec{c}(\vec{x}) = [\rho\ \sigma^2_{lfm}\ \vec{\psi}_{v,lfm}(\mathbf{X}_{s,lf}, \vec{x}), \rho^2\ \sigma^2_{lfm}\ \vec{\psi}_{v,lfm}(\mathbf{X}_{s,hf}, \vec{x}) + \sigma^2_d\ \vec{\psi}_{v,d}(\mathbf{X}_{s,hf}, \vec{x})]^T \tag{2.23}$$

and the prediction is then made with:

$$\hat{y} = \hat{\mu}_d + \vec{c}\,\mathbf{C}^{-1}(\vec{Y}_{s,hf} - \vec{1}\mu_d) \tag{2.24}$$

where $\vec{1}$ is a vector with length of all samples containg ones. The model error is computed in a similar fashion to regular Kriging by:

$$\hat{s}(\vec{x})^2 = \rho^2\hat{\sigma}^2_{lfm} + \hat{\sigma}^2_d - \vec{c}^T\mathbf{C}^{-1}\vec{c} + \frac{1 - \vec{1}^T\mathbf{C}^{-1}\vec{c}}{\vec{1}^T\mathbf{C}^{-1}\vec{1}} \tag{2.25}$$

Hierarchical Kriging

In opposition to Co-Kriging, Hierarchical Kriging does not directly correlate low and high-fidelity samples. Hierarchical Kriging is proposed by Han and Görtz [106], which shares similarities with Kriging with a drift, see Wackernagel [132]. The concept is to exchange the trend of a regular (universal) Kriging model with the trend of another low-fidelity surrogate model. The version by Han and Görtz is posed as follows:

$$f_{trend}(\vec{x}) = \rho\hat{y}_{lfm}(\vec{x}) \tag{2.26}$$

with ρ being a scaling constant. Xiong et al. [107] add an additional constant offset γ to the trend:

$$f_{trend}(\vec{x}) = \rho\hat{y}_{lfm}(\vec{x}) + \gamma \tag{2.27}$$

In this work the following formulation proved to yield the best results:

$$f_{trend}(\vec{x}) = \rho\hat{y}_{lfm}(\vec{x}) + \sum_{k}^{d}(\beta_k x_k) \tag{2.28}$$

which yields a mix of first order polynomial model and the low-fidelity. The latter allows the Kriging model to be more flexible, as it can shift the location of the optimum slightly.

The to be determined coefficients of the Hierarchical Kriging model are handled similarly to universal Kriging, thus the extension to Hierarchical Kriging from an existing universal Kriging framework is made quickly. The regression vector is rewritten as $\vec{f} = (\hat{y}_{lfm}(\vec{x}), x_1, ..., x_d)$.

The hyper parameters are also tuned through the optimization of the maximum likelihood function, while the values of $\hat{s}$, $P[I(\vec{x})]$ and $E[I(\vec{x})]$ are computed similar to universal Kriging. Care should be taken, if multiple goal functions are considered. The trend is dependent on the different low-fidelity goal functions, which is not the case for universal Kriging, where the trend is only built from the location of the samples $\mathbf{X}_s$.

Putting it all together, the model is written in a similar form as Equation (2.4):

$$\hat{y} = \rho\hat{y}_{lfm}(\vec{x}) + \sum_{k}^{d}(\beta_k x_k) + \psi_{v,hfm} \cdot \mathbf{\Psi}_{hfm}^{-1}(\mathbf{X}_{s,hf}) \cdot (\vec{Y}_{s,hf} - \mathbf{F} \cdot \vec{\beta}) \tag{2.29}$$

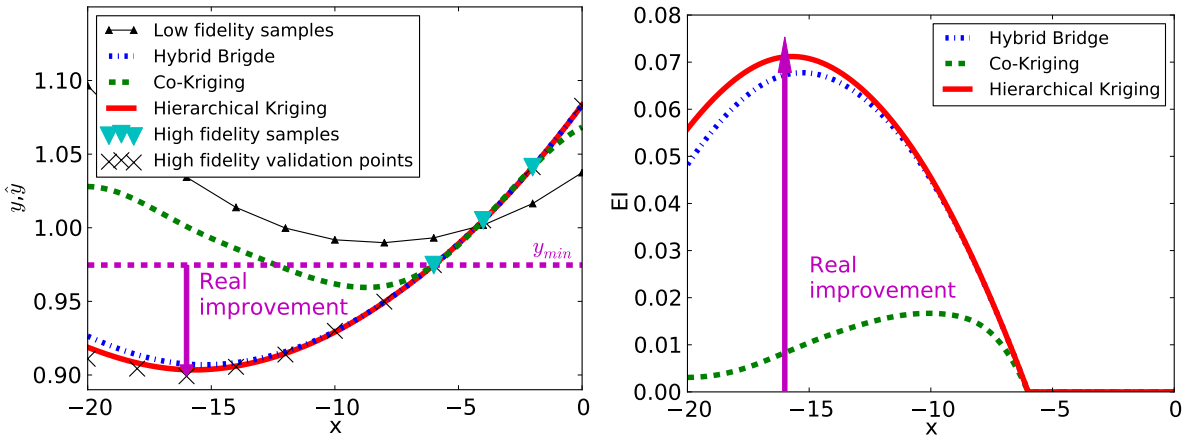
which exactly interpolates the high-fidelity data, unless the noise constant Λ is added to the correlation matrix $\mathbf{\Psi}$ of the variable-fidelity model.

2.3.6 Selecting the Variable-Fidelity Method

From a theoretical point of view, Hierarchical Kriging is better than Co-Kriging, because Co-Kriging has two disadvantages; first, the larger correlation matrix with size $(n_{lf} + n_{hf})^2$ consumes more memory and contains cross-correlations between low- and high-fidelity points, which may lead to off-diagonal elements being one. The latter is problematic when inverting/decomposing the correlation matrix, which has to be done every time a new sample is added to the model, independent of the fidelity it belongs to. The second disadvantage is that Co-Kriging yields an EI function, which takes into account the model error of the low-fidelity. Thus, at points where the low-fidelity samples exist, the variable-fidelity model error may decrease unproportionally high and false predictions of $\hat{s}$, PI, EI and DC are possible.

The major difference between Hybrid Bridges and Hierarchical Kriging is that the multiplicative term of the Hybrid Bridges in front of the low-fidelity model is determined outside the Kriging process, while Hierarchical Kriging does this internally. As for the computational effort for the creation of correlation matrices, this reduces to $n_{lf}^2 + n_{hf}^2$ for the initial setup for Hybrid Bridges as well Hierarchical Kriging. Yet, if the low-fidelity model is not touched anymore during the optimization cycle, only the high-fidelity correlation matrix of the size n_{hf}^2 needs to be decomposed.

A comparison of Hybrid Bridges, Co-Kriging and Hierarchical Kriging is displayed in Figure 2.6 for the function approximation on the left and the EI function on the right. The test data is abstracted from the parametric study in hover for the twist parameter found in Section 3.3. The low-fidelity is given by the finite state model and Euler simulations pose the high-fidelity. The low-fidelity surrogate model is created with eleven low-fidelity points, while the high-fidelity is only build with three data points. The high-fidelity validation data set consists also of eleven points. In all three test



(a) function approximation

(b) expected improvement

Figure 2.6: Comparison of Hybrid Bridges with Co- and Hierarchical Kriging

cases, the surrogate models are allowed to adjust the noise constant Λ as well as the hyper parameter θ. These hyper parameters are sought in a uniformly sampled space with 16 samples and a local optimizer deployed to adjust these parameters locally.

The function approximations of the Hybrid Bridge and the Hierarchical Kriging are in good agreement with the validation samples seen in Figure 2.6(a), but the Co-Kriging predictor tends to resemble the low-fidelity more, in particular the location of the minimum. This is also reflected by the EI. While the EI of the Hybrid Bridge and Hierarchical Kriging match closely the real possible improvement of the high-fidelity samples of about 0.075 in Figure 2.6(b), the EI of Co-Kriging is smaller and also points towards the optimum of the low-fidelity data. Fortunately, the tuning process for the Co-Kriging avoided the strong effect of the low-fidelity samples on the EI function in this case, as no bumps are noticed at the low-fidelity locations, contrary to the expectation. Thus, the major offset from the EI comes from the stronger deviation of the goal function prediction, which orients itself more closely to the low-fidelity model than the Hybrid Bridge or Hierarchical Kriging.

Since Hierarchical Kriging features slightly better results and has reduced overhead due to its compact formulation over Hybrid Bridges, it is decided to use Hierarchical Kriging from now on.

2.4 Numerical Optimization

The task of a numerical optimization is to improve a system from its initial state to a better state defined by the metrics of goal functions. The key idea is to obtain this improved state systematically and efficiently. Formulated differently, for a function, which returns the significant metrics of the system to be improved, with a certain number of input parameters, the problem is to find:

$$\vec{F}(\vec{x}) \rightarrow \min \tag{2.30}$$

Where $\vec{F}$ is a set of goal functions and $\vec{x}$ the design vector. The simple case, where $\vec{F}$ becomes a scalar, is called a single-objective optimization, while problems with more than one goal function are called multi-objective.

Another aspect of numerical optimization are constraints. They can either be treated implicitly or explicitly. There are two types of constraints:

$$\vec{h}(\vec{x}) = \vec{h}_{point} \tag{2.31}$$

$$\vec{g}(\vec{x}) < \vec{g}_{max} \tag{2.32}$$

The first constraint vector $\vec{h}(\vec{x})$ represents equality constraints. For example the constraint could be that a trimmed aircraft is considered in the optimization for which the forces and moments are equal to zero. This type of constraint may either be handled implicitly by the simulation code, or explicitly, the optimization algorithm finds the equilibrium of forces. The second constraint vector $\vec{g}(\vec{x})$ being a set of inequality constraints, could be a test whether the maximum stress on a structure is exceeded or not.

Equally for goal functions and constraints, a maximization problem can be transformed to a minimization problem by negating the function value to the optimization algorithm. In the literature, it is often mentioned that the reciprocal value of the function can be optimized, but this leads to unsmooth functions. For example $F = x$ is smooth and differentiable, while $F = \frac{1}{x}$ is not defined for $x = 0$. Therefore, the negation of the goal function or constraint is favored, unless there is strong requirement for the inverse.

When it comes to multi-objective problems, there are two major approaches available. The first approach is the application of different weights to the individual goal functions and it returns the sum of them to the optimizer:

$$y_{ws}(\vec{x}) = \sum_{i}^{g} w_i y_i(\vec{x}) \tag{2.33}$$

with w_i the individually chosen weights and y_i the according goal functions. The advantage of this approach is the simplicity. The multi-objective problem is reformulated as a single-objective problem. However, depending on the weights, one of the goal functions may be favored more than the others. The second approach seeks design points, which are non-dominated. A design is non-dominated, if no other point exists, which surpasses this design in all goal functions. This is checked by going through the list of considered points with:

$$\text{if } [min(\vec{y}_j - \vec{y}_i) > 0] \text{ then point } i \text{ is dominated by point } j \tag{2.34}$$

with $\vec{y}$ the vector of goal functions. Going through all goal functions for a set of points and gathering points that are not dominated by any other point, then this set of points is called Pareto front established by this Pareto criterion.

For clarification, Figure 2.7 shows how points a, b and c are part of the Pareto front, while point d is not. Point a is the best point for goal function 1, while point c is the best for goal function 2. These two points are then also the anchor points of the Pareto front. Point b is a non-dominated design, as neither point a, nor point c improves both goal functions of this point. In contrast, point d is dominated by point b in both goal functions and is thus not part of the Pareto front. However, if point b would not exist, point d would belong to the Pareto front, too.

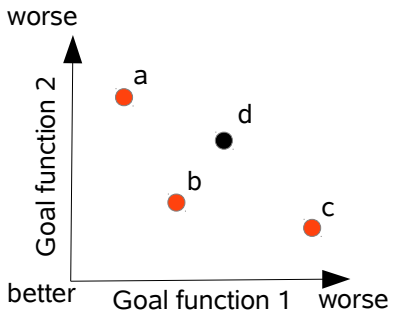


Figure 2.7: Example Pareto front. Red points are part of the Pareto front

The beauty of a Pareto front is that it directly allows the engineer to see the trade-off between the various goal functions. The difficulty lies in recovering the Pareto front, which in general requires a lot more function evaluations than a weighted sum approach.

2.4.1 Hybrid Optimization Strategy (Single-Objective)

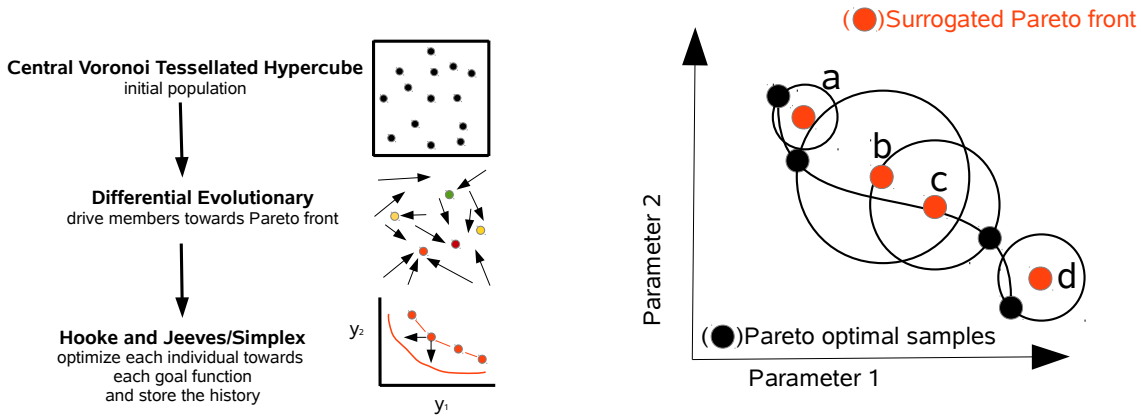
Since surrogate models are evaluated quickly, the search algorithm can be more costly in contrast to direct optimization, if the overall accuracy is improved. There are a large number of optimization algorithms, and most of them are customized to a specific problem. As the expected improvement is not a continuous function, it is wise to use a global optimizer first to spot the region of interest. To improve the accuracy of the optimum, a local optimization algorithm is started to home in on the optimum. A gradient based optimizer is considered unsuitable for this task for two reasons. Firstly, the analytical gradient of Kriging surrogate model is as costly as obtaining the gradient through finite differencing. Secondly, the aforementioned discontinuity of the expected improvement is impractical for gradient based optimizers.

Two hybrid optimization strategies are used in this thesis. The first strategy is tailored for lower, here less than five independent variables, and the other strategy for higher dimensional optimization problems. For low dimensional problems, a brute force methodology guarantees a deterministic result:

Strategy 1:

1. Perform a Full Factorial DoE with 8 samples in each spatial direction
2. Start the Hooke and Jeeves [112] pattern search method from the best point found from the DoE to locally refine.

The Hooke and Jeeves algorithm is a very robust local pattern search method, which spawns a multi-dimensional cross in the design space. It moves into the direction of the smallest goal function value of each arm of the cross. If no more improvement is found, the cross is shrunk and the search continues. The algorithm is explained in more detail in Section A.1.2. However, even with surrogate models this brute force strategy is be prohibitively expensive for more than four parameters, thus a different hybrid optimization strategy is developed:



(a) Step 1: Hybrid/multi-objective optimization strategy

(b) Step 2: The selection process of multiple-objective infill samples

Figure 2.8: Sketch of the multiple objective infill method

Strategy 2:

1. Build a LHS that is CVT optimized as an initial population
2. Start the Differential Evolutionary (DE) algorithm [116] from this initial population
3. Start the Simplex pattern search method [113] from this optimum to locally refine.

Starting with the CVT based DoE yields a more diverse initial population for the DE algorithm. The DE algorithm is based on the idea of evolution, where only the fittest individuals survive. In the context of single-objective optimization, this is the optimum. The evolution, mutation and cross-over of genes, is achieved through random vector operations for the DE. The here applied differential evolutionary algorithm is an extension of the original formulation by Storn and Price [116]. It features better local convergence and is developed by Das et al. [133]. The algorithm is explained in greater detail in Section A.1.2. The simplex pattern search method is a more efficient version of the Hooke and Jeeves method, which does not consist of a multi-dimensional cross, but of a simplex, thus reducing the total number of points in the pattern. The simplex algorithm also features more movement operations than the Hooke and Jeeves algorithm, which are described in Section A.1.2.

2.4.2 Multi-Objective Infill Method

An approach that finds the Pareto front is constructed and sketched in Figure 2.8(a). It exhibits the same hybrid optimization strategy elements as **Strategy 2** defined in the previous section, yet it is modified for multi-objective problems.

The approach to use a hypercube as the initial population is the same, and afterwards starting the differential evolutionary algorithm. The original formulation of this algorithm is modified to allow the treatment of multi-objective problems by including the non-dominated sorting genetic algorithm II (NSGA-II) by Deb et al. [91]. The NSGA-II is explained in more detail in Section A.1.2. The NSGA-II method basically helps the optimizer to decide whether one individual is more interesting than the other and thus

survives in the evolutionary process. For a single-objective problem this reduces to one goal function value being smaller than the other one.

In this work, a local refinement strategy using the Hooke and Jeeves algorithm is developed to further improve the quality of the Pareto frontier after the run of the differential evolutionary algorithm. This is achieved by starting the local search method at each individual of the differential evolutionary algorithm and then moving towards each goal function. The history of the searched points is kept and afterwards checked against the Pareto optimality. In order to not miss any points, the step size of the local search algorithm is set to $\epsilon = 5 \cdot 10^{-3}$ in the unit design space.

Another contribution of this work is the selection methodology of infill points based upon the rigorously resolved Pareto front obtained from the surrogate model. For a sequential infill, the surrogated Pareto optimal point with the highest minimal distance from a sampled point is chosen. For a parallel infill the number of points is first reduced, if necessary, to the requested number of parallel infill samples. This is done by spawning a virtual radius around each point and checking that no other point is within this radius. This radius is increased or decreased until a number of points close to, but larger than, the number of points requested exists. Then the set is sorted from the highest to the lowest minimal distance to the next existing sample and truncated to the requested number of samples. Through this distance sorting and only selecting points with a great distance to the already sampled points, this infill strategy gains an exploratory part, which ensures that the virtual Pareto front does not get stuck in artificial or real minima. Exemplary, the selection process is sketched in Figure 2.8(b). The black dots represent the Pareto optimal set of simulated samples, while the orange dots represent the Pareto optimal points found in the surrogate models by the multi-objective search strategy. The black circles around the surrogated points mark the minimal distances to the next simulated sample location. If a single point is requested, then point b would be chosen, as it features the greatest distance from the next sample. If three points are requested, points a, b and d would be chosen. Point c features a higher distance from the next Pareto optimal sample then point d, however, as it is too close to point b, it is not considered as an infill sample. It is expected that point d brings in more new information than point c does, if point b is also sampled.

2.5 Optimization Framework

The general outline of the variable-fidelity based optimization has been presented in Figure 2.1 in Section 2.1. A major part of this work has been the creation of this framework and thus in this section the implementation specific details are presented. Certain technical issues are addressed, which may not occur in an ideal setting, but on practical terms occur often and lead to otherwise avoidable failure. Also, potential differences to other frameworks may be found here for comparison.

2.5.1 Parameter and Response Scaling

All discussions are based upon a hyper unit cube meaning that all parameters are treated in a $[0, 1]$ space. Later on, the response values of the simulation are also scaled between $[0, 1]$ for the surrogate model. This scaling has two purposes: first, numerical instabilities arising from very differently ranged parameters shall be eliminated, and

second, the impact of each design variable is weighted, so that the importance of each parameter matches the actual design space. Scaling the responses becomes particularly helpful in the context of variable-fidelity methods.

2.5.2 Avoidance of Re-computation

When using the noise constant λ to stabilize the Kriging models, as well as regress data points, the expected improvement function will not completely be zero at already computed points as the predicted error will also not be zero. In order to avoid to re-compute an already evaluated point, an additional measure is introduced to avoid infilling an existing point for single-objective problems. If the chosen infill point already exists in the surrogate model, the infill metric is cycled through model error, expected improvement and function value. If all of these metrics still try to predict a given point, a random point will be added, which is not in the surrogate model.

The point existence check is simply based upon the 2-norm of the difference between selected infill point and existing data points. Should this value drop below a user specified value δ, here 0.5% in unit space, the point is considered as existent in the model. This fairly coarse tolerance value also guaranties that the Kriging model does not become unstable, as values too close to each other cause high correlations on the off-diagonal entries of the correlation matrix, making it difficult to invert. This minimum distance measure has also been proposed by Horowitz et al. [92] to keep a sane surrogate model.

For multi-objective problems, the safe distance check inherently exists by the infill strategy previously explained in Section 2.4.2.

2.5.3 Design Space Shrinking

Li [93] demonstrates that shrinking the design space in terms of parameter bounds results in faster convergence of the overall surrogate based optimization process. Leusink et al. [127] reduce the design space after a preliminary low-fidelity search to speed up the convergence of their genetic high-fidelity optimization. A related mechanism is implemented for accelerating the variable-fidelity optimization. The design space is reduced after the low-fidelity sampling based upon the assumption that the best low-fidelity point is in vicinity of the high-fidelity optimum. While the high-fidelity design of experiments is still performed in the whole design space to ensure a good prediction capability at the new boundaries, the update cycle later on is restricted to a design space that centers on the low-fidelity optimum. The distance ϵ from this optimal point is set ± 0.25 Euclidean norm in the unit design space for single objective optimizations. For example, if one optimal parameter of the low-fidelity is 0.6, the search range will be limited to $[0.35, 0.85]$ in comparison to the original range of $[0, 1]$. In order to maintain a certain flexibility, new ranges which exceed the initial boundaries are adjusted to fit into the original design space. The difference overshooting the original design space is added to the opposite side of the range. Let the low-fidelity optimum be at 0.9 with an $\epsilon = \pm 0.15$, then the new search range is not set to $[0.75, 1.05]$ but to $[0.7, 1.0]$.

In lower dimensional spaces, the decrease of the search space is relatively small; however with increasing number of parameters, the searched volume is minimized significantly. If the design space volume is specified as the product of the edge length of

the hypercube, then the original volume would be 1 for any number of parameters. Applying an $\epsilon = \pm 0.25$, the new edge length would be 0.5. For one parameter, the hyper volume would be 0.5, for four parameters $\frac{1}{2}^4 = 0.125$ and for ten parameters $\frac{1}{1024} \approx 0.001$. The smaller volume helps the optimization process significantly to home in on the optimum. Though, if the high-fidelity optimum is not within the vicinity of the low-fidelity one, the chance of finding the true high-fidelity optimum is nullified.

2.5.4 Treatment of Unsuccessful Computations

A great and general problem within numerical optimization is the treatment of unsuccessful computations. This ranges from failures in the grid generation, due to an unfortunate design parameter choice leading to skewed surfaces, non- or badly-converging CFD computations, and unphysical answers of the simulation system. The latter is particularly the case when optimizing rotor blades. A configuration might fail to generate the demanded thrust or does not find an aero-elastic equilibrium, which the comprehensive code returns as an unsuccessful trim.

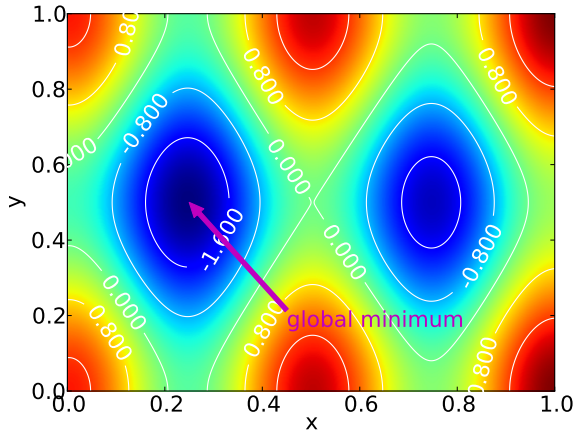
There are various ways on how to deal with this problem. Assigning a penalty value might work well, if the function is directly optimized with a non-gradient based optimization algorithm. Yet feeding penalty values to a surrogate model renders it useless, due to false predictions. A common approach is proposed by Forrester [95], who places the function value from the surrogate model into the set of samples for the surrogate model generation. This way, the location of the failed computation is not called again, while not tainting the surrogate model too much.

A new route is developed in this work. An additional surrogate model is constructed, which is built with the sample locations of the simulated points. The function value is either 0 for a successful computation or $1 + \delta$ for a failed computation. This surrogate model then predicts the probability of a successful run, which is considered if the function value is below 1. δ is thus a small number to be added to additionally safe guard a small region around a failed point. In this work, δ is chosen as 0.1 to ensure this certain distance from unsuccessful configurations. This error mapping surrogate model is then handled like any other surrogate model for constraint functions and the expected improvement is simply multiplied with the probability that this point will not fail. If two or more goal functions are requested at a point where failure is predicted, this point is marked as invalid and not further addressed by the optimizer. This additional surrogate model is referred to as 'crashmap' as it allows visualizing the regions where the novel configurations fail due to an unachievable trim. The strength of this approach is that no unfeasible goal function values are added to the goal function surrogates as no information from unsuccessful configurations is inserted. Only the crashmap carries this knowledge and avoids evaluating unfeasible designs in the future.

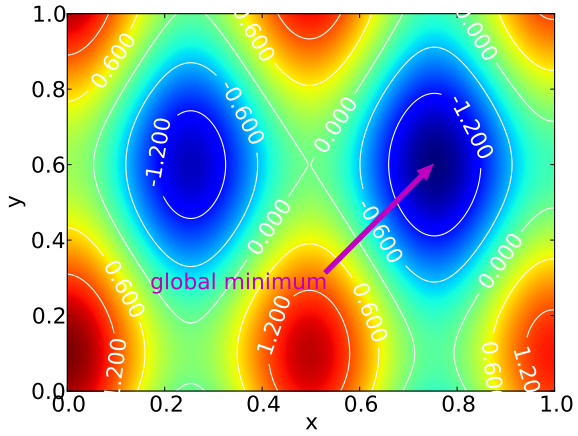
2.6 Example Problem

A sandbox problem is created to first compare the single and variable-fidelity optimization. This has two advantages: first, the truth function is computed quickly, and second, the comprehension of a 2D analytical function is by far easier than a higher dimensional rotor blade problem.

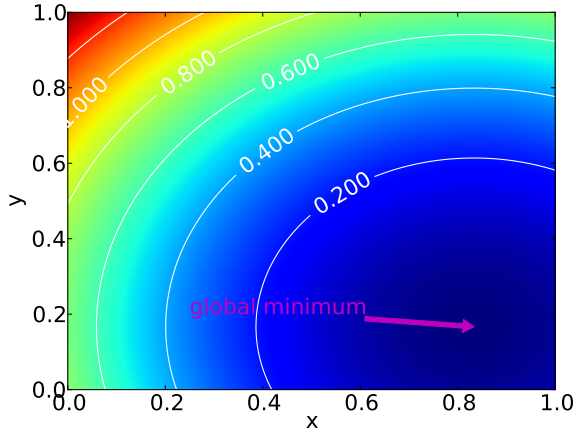
2.6.1 Test Function



(a) High-fidelity function



(b) Low-fidelity function A



(c) Low-fidelity function B

Figure 2.9: Synthetic test functions

The test function is made of two cosine terms super-positioned with a linear term:

$$\begin{aligned} z_{hfm}(x,y) &= cos(4\pi x) \\ &+ cos(2\pi y) \\ &+ \tfrac{1}{2}\cdot(x-\tfrac{1}{2}) \end{aligned} \tag{2.35}$$

The high-fidelity function is pictured in Figure 2.9(a) for the unit design space and features two minima, one at $(x,y) = (\frac{1}{4},\frac{1}{2})$ with a value of $z = -2.125$ and $(x,y) = (\frac{3}{4},\frac{1}{2})$ with a value of $z = -1.875$. In order to test the robustness of the process of the variable-fidelity approach, two low-fidelity functions are created. The first low-fidelity function, designated A, is a slight modification to the original high-fidelity function:

$$\begin{aligned} z_{lf,1}(x,y) = \tfrac{4}{5}\cdot[\ & cos(4\pi & x &) \\ &+ cos(2\pi & (y-\tfrac{1}{10}) &) \\ &-\tfrac{1}{2} & \cdot(x-\tfrac{1}{2}) & \ ] \end{aligned} \tag{2.36}$$

which flips the global optimum to the other side and shifts both minima upwards, therefore decreasing the absolute value. The minima of the function of Equation (2.36) within the unit design space are then:

- $(x, y) = (\frac{1}{4}, 0)$ with a value of $z = -0.0528$
- $(x, y) = (\frac{3}{4}, 0)$ with a value of $z = -0.2528$
- $(x, y) = (\frac{1}{4}, \frac{3}{5})$ with a value of $z = -1.50$
- $(x, y) = (\frac{3}{4}, \frac{3}{5})$ with a value of $z = -1.70$

which are drawn in Figure 2.9(b). The goal is to test if the approach is globally accurate, even if the optimal position of the low-fidelity lies somewhere else.

The second low-fidelity function, designated B, is purposely chosen to have nothing in common with the original high-fidelity function. In theory, the surrogate based optimization process should still be able to converge. The error term of the Hierarchical Kriging will correct the offset after many high-fidelity samples are evaluated. However, it should require more samples in total to achieve this correction. The function is parabolic and describes a circle:

$$z_{lf,2}(x, y) = (x - \frac{5}{6})^2 + (y - \frac{1}{6})^2 \tag{2.37}$$

This function has only one optimum at $(x, y) = (\frac{5}{6}, \frac{1}{6})$ with $z = 0.0$ sketched in Figure 2.9(c).

2.6.2 Setup

In order to benchmark single- versus variable-fidelity, a set of different test cases is set up and visualized in Figure 2.10. On top of this flowchart the single-fidelity test cases are depicted, in the middle the variable-fidelity cases with low-fidelity function A, and on the bottom the variable-fidelity setup with low-fidelity function B. While the variable-fidelity setups are the same for both low-fidelity functions, the single-fidelity test cases do not acquire any data from the low-fidelity. The methodology is the one explained in section Section 2.1. The naming convention is as follows; first **SF** or **VF** distinguishes the case, whether it is single- or variable-fidelity, second, the underlying low-fidelity (**LF A** or **LF B**) is specified, which is omitted for the single-fidelity, and third, the type of high-fidelity sampling. **Point** refers to a design of experiments consisting of a single point, which is a reference candidate for the single-fidelity. For the variable-fidelity setup, the **Point** sample is the best found low-fidelity configuration. Numbers, such as **4** or **10** denote the number of random samples in the design of experiments. The number of high-fidelity updates is kept the same for all benchmarking tests performed in this thesis and is set to 50 for this test. Reason for this is to maintain comparability between the single- and variable-fidelity tests. Picking out the case **VF LF A Point**, the optimization run is organized by running a random design of experiments on the low-fidelity function A with 40 samples and generating a low-fidelity surrogate model from it. The optimum of this low-fidelity surrogate is sampled as the only point of the high-fidelity design of experiments. The optimization process then proceeds by searching for the point of the highest expected improvement and sampling this point with the high-fidelity function. This is repeated until either the design confidence of more than 90% is achieved in three consecutive cycles or until 50 samples have been evaluated. Note: the design confidence is a desirable stopping criterion as it gives the user a feedback on how trustworthy the current optimum is. Therefore,

the resources required to reach the design confidence are additionally measured as it reflects the resources one would invest to be sure to have reached the global optimum.

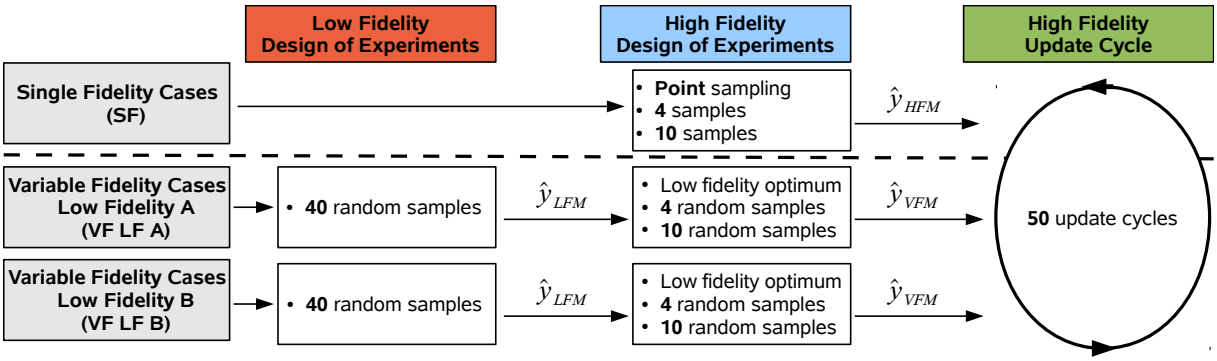
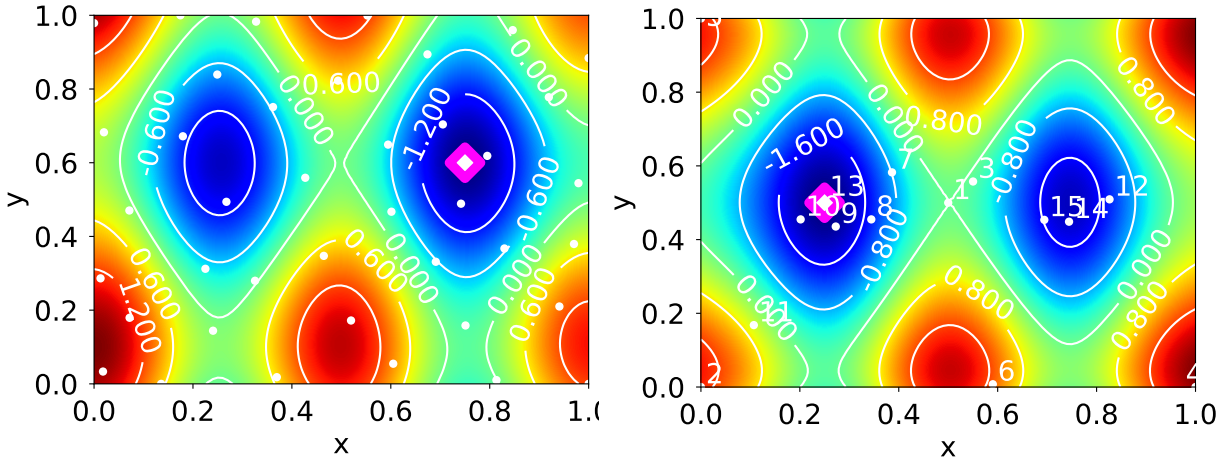


Figure 2.10: Test setup for the demonstration test case of the synthetic goal function

For the single-fidelity method, universal Kriging with a second order polynomial trend without mixed polynomials is employed. The order is reduced if not enough points exist yet, for example when starting from a single point. For the variable-fidelity, the Hierarchical Kriging approach is corrected with a first order bridge. The bridge is neglected if insufficient amount of points exists. The low-fidelity model adheres to the same features as the single-fidelity. The hyper parameters as well as the expected improvement are found using optimization **strategy 1** from Section 2.4. The bounds for θ of $[10^{-4}, 10^{2}]$ are the most favorable from initial tests, while for λ the search range is set between $[10^{-15}, 10^{-3}]$. The exponent p of the Gaussian like Kriging radial basis function is fixed to 2. The threshold for the design confidence is set to 0.01, which stops the optimizer if the best sampled goal function is $-2.11\mathbf{5}$ and the surrogate does not predict a point having a better goal function than $-2.12\mathbf{5}$ with absolute certainty.

2.6.3 Results

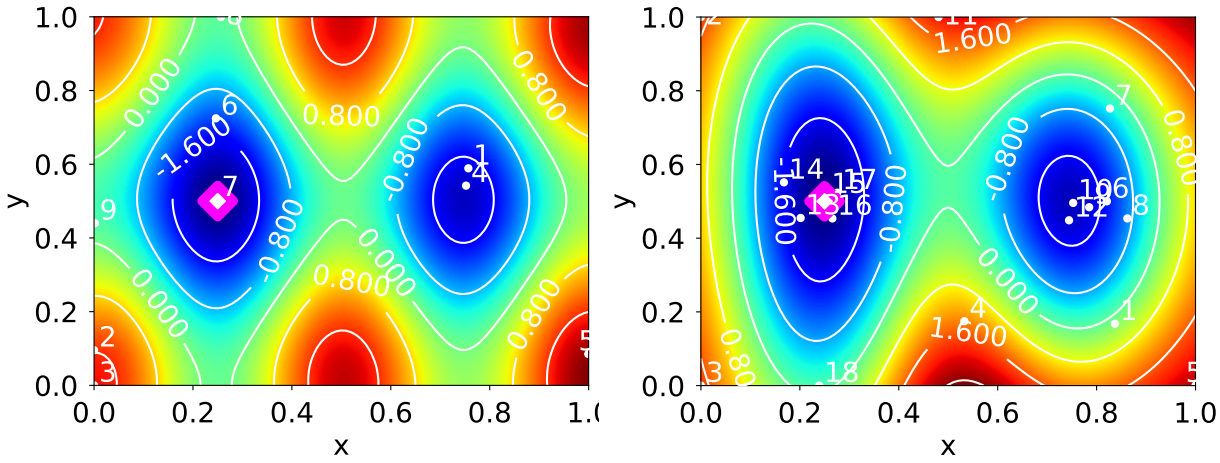
To give an overview of the quality with which the surrogate models approximate the analytical function, the low-fidelity surrogate as well as the final surrogates of the single point optimizations are plotted in Figure 2.11(a) to Figure 2.12(b). Starting with Figure 2.11(a), the low-fidelity surrogate model resembles the analytical low-fidelity function (A) pictured in Figure 2.9(b) closely. The white points in the approximation plot show the sample location of the 40 samples from the random design of experiments. The magenta diamond ◊ marks the optimum of each function. The surrogate model of the single point single-fidelity (**SF Point**) optimization case is depicted in Figure 2.11(b). The features of the high-fidelity function (Figure 2.9(a)) are well replicated by the surrogate model and both minima are identified with three samples each in their vicinity. Looking at Figure 2.12(a), where the surrogate model of case **VF LF A Point** is pictured, the surrogate is in very close agreement with the analytical function, with even less samples evaluated than for the **SF Point** case. A few samples are also positioned at the boundaries, but only three samples suffice to spot and identify the two minima correctly. The case is different with **VF LF B Point**. The minima are both found, see Figure 2.12(b), but further away from the minima the approximation degrades. In general, more samples are placed around the minima to correct the misleading trend function, LF B.



(a) Surrogated Function LF A

(b) Single-fidelity and single point sampling

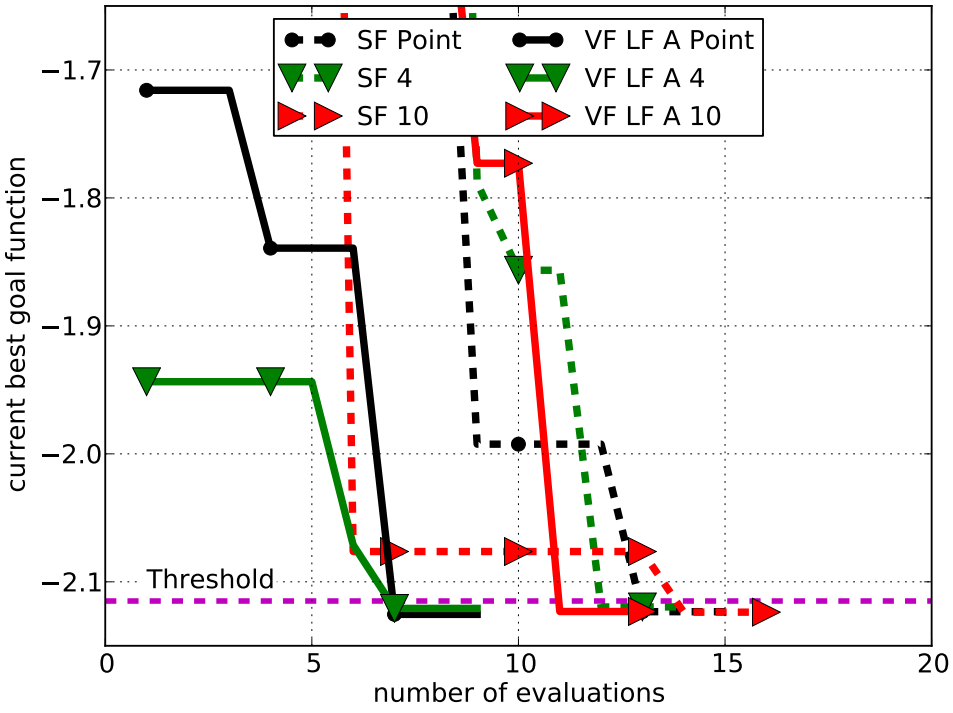
Figure 2.11: Single- and low-fidelity surrogate models after optimization of synthetic test case



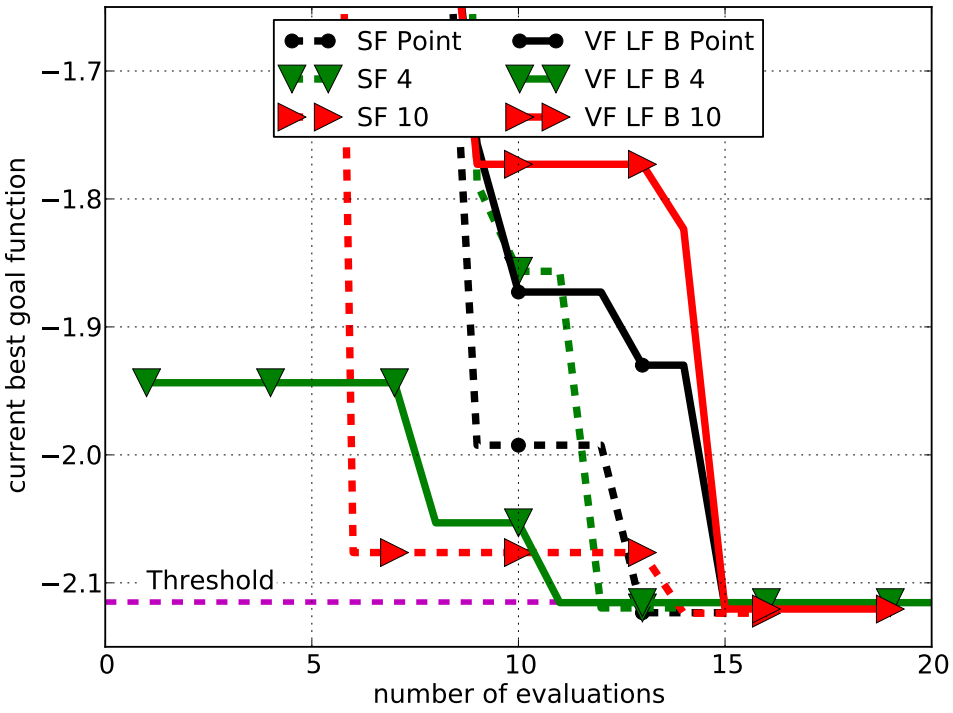
(a) Variable-fidelity with single point DoE and LF A

(b) Variable-fidelity with single point DoE and LF B

Figure 2.12: Surrogate models after optimization of synthetic test case



(a) SF and VF LF A



(b) SF and VF LF B

Figure 2.13: Convergence of single- (SF) to variable-fidelity (VF) optimization of synthetic test case

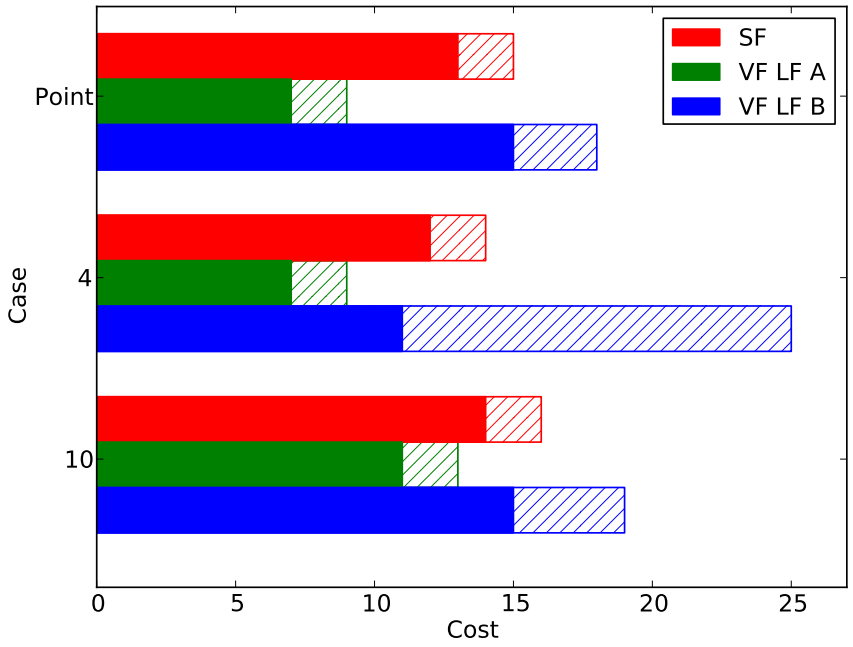


Figure 2.14: Cost to reach either threshold (filled area) or reaching design confidence (DC) criterion (hatched area). SF = single fidelity, VF = variable fidelity

From this initial observation, the performances of the individual optimizations are benchmarked. All optimizations managed to converge towards the global optimum, independently of the quality of their respective surrogate models. Comparing the three single fidelity runs with the variable-fidelity runs that are augmented with low-fidelity function (A), Figure 2.13(a) graphs the current best goal function over the number of high-fidelity evaluations. The single-fidelity is presented by dashed lines (--), whereas the variable-fidelity is visualized by the continuous lines (—). The threshold is drawn by the magenta dashed line to indicate when the surrogate based optimization process reached the optimum within the margin of 0.01. From Figure 2.13(a), the black and green continuous lines reach the threshold with seven samples, thus the variable-fidelity process is quicker for these design of experiments (**Point** and **4**) than their single-fidelity counterparts. The **VF LF A 10** case finishes two samples before the **SF 4** case. The variable-fidelity process with low-fidelity function (B) is less efficient in contrast. In Figure 2.13(b) it is seen that the **SF Point** and **SF 10** cases are faster than the **VF LF B Point** and **VF LF B 10** cases. Only the **VF LF B 4** is faster than its single-fidelity counterpart, but continues to sample more points than the other cases after having found the optimum. The reason for this is that the design confidence is reached after a total number of 26 samples for this case.

In Figure 2.14, the merits of each optimization are pictured. The bar chart has filled and hatched portions of the bars. The filled part marks the cost to find the optimum in terms of high-fidelity computations, while the hatched area indicates the additional cost to obtain a design confidence over 90% in three consecutive update cycles. In theory, a designer could finish after reaching the threshold and this would be the appropriate metric for comparison. In practice the designer does not know when the optimization is finished and the design confidence is a metric which yields a valuable stopping criterion. Thus, the resources required for either metric are discussed, as the

resources required for reaching the threshold reflect the theoretical cost, while the resources required to ensure the design confidence reflect the cost one would invest for an unknown goal function.

For all three types of design of experiments of **VF LF A**, the overall costs to reach the optimum are less than the other single- or variable-fidelity tests with low-fidelity function (B). The design-confidence is reached within two more samples for the **VF LF A** cases. The single-fidelity cases **SF Point** and **SF 10** are faster than the equivalent **VF LF B** cases, and the **SF 4** case is one sample slower than the **VF 4** case. In terms of reaching the design confidence, the single-fidelity is superior in all three cases over the variable-fidelity with low-fidelity function (B). Quantifying the efficiency of the best **VF LF A** and **VF LF B** cases relative to the single-fidelity, **VF LF A 4** saves 41.7% high-fidelity samples to reach the optimum and 35.7% to reach the design confidence. **VF LF B Point** increases the cost for the design confidence by 28.6% in contrast to **SF 4** and the cost for finding the optimum is smaller by 8.3% with **VF LF B 4**.

2.6.4 Conclusions

From this synthetic test problem, a few insights about the variable-fidelity approach are gained.

- When correctly set up, the variable-fidelity approach reduces the number of high-fidelity samples, while generating a more accurate final surrogate model.
- It is strongly recommended to have a valid low-fidelity function, as an uncorrelated low-fidelity has a severe impact on the performance and success of the optimization.
- A good initial sampling with a well-chosen size is crucial to keep the overall run time at a minimum.

This implies two things for the low-fidelity model. On the one hand the physical function has to be similar to the high-fidelity function and on the other hand the surrogate model built from it has to reproduce the design space similarly. If any of these two conditions fails, then the suitability of the variable-fidelity optimization approach is doomed to be less efficient than single-fidelity optimizations.

3 Investigation and Selection of Aerodynamic Models

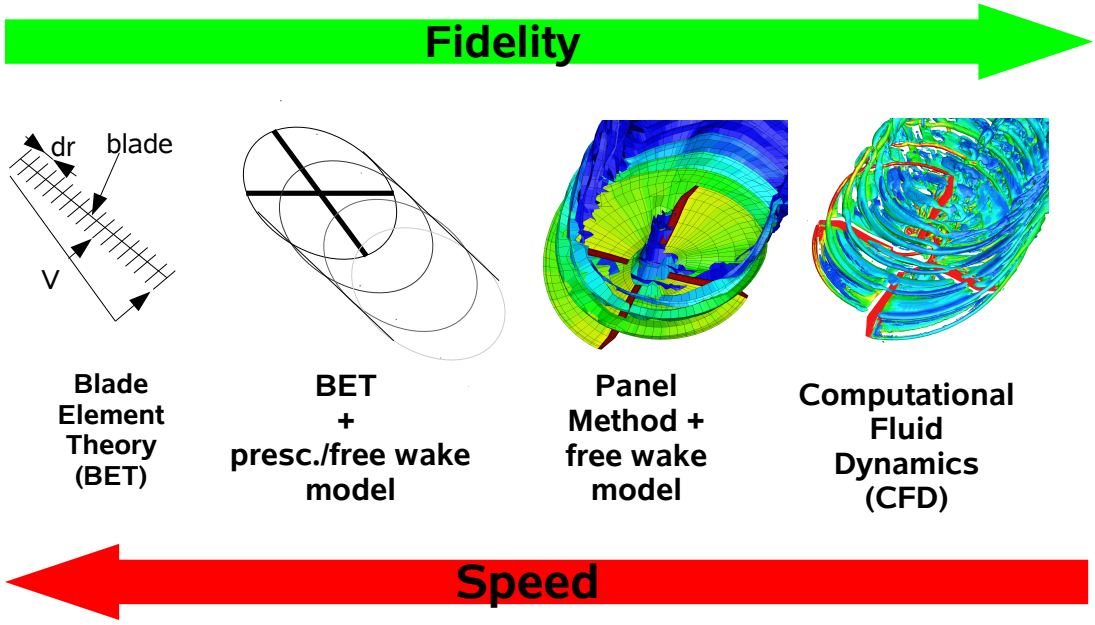


Figure 3.1: Sketch of methods for rotor aerodynamics

There is a wide range of aerodynamic models available for the prediction of performance of helicopter rotor blades [28]. The fidelity varies greatly and with it the computational effort required, schematically sketched in Figure 3.1. Simple methods for computing rotor blades may consist of the blade element theory (BET), which can be enhanced using wake models. Inviscid surface methods such as panel methods coupled with a free wake model resolve the general flow field better, but viscous computational fluid dynamics (CFD) solutions feature the high end of fidelities available. On the one hand, the fidelity increases, while on the other hand the computational effort grows exponentially. Goal of this chapter is to come up with a choice of aerodynamic models suitable for the variable-fidelity process mentioned in Chapter 2. Therefore, the driving physical effects of each rotor blade planform parameter are discussed first, followed by a theoretical review of the given aerodynamic models and their abilities to model these physical aspects. To further quantify the error of these methods, a sensitivity study as well as a genetic optimization of a rotor blade is performed to identify the effects of the different representation. The chapter is closed with a choice of fidelities to be investigated more deeply in the variable-fidelity process in Chapter 4. It is noted that the focus is laid upon correctly modeling the rotor performance in dependency of the geometric properties in the flight conditions of hover and forward flight, other metrics are of no concern here.

3.1 Planform Parameters and their Influence

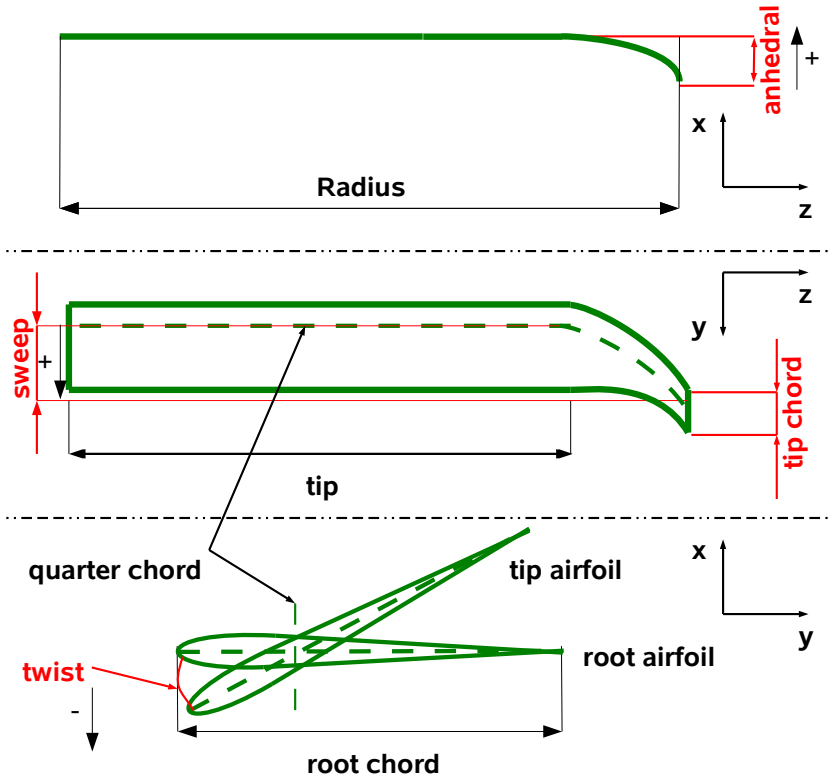


Figure 3.2: Sketch of 7A parameterization. The twist is varied linearly over the blade span, while anhedral, chord and sweep are varied with a quadratic function starting from the blade tip location.

This section discusses the four major planform parameter groups available to helicopter rotors and their driving physical effects observed. The goal is to refine the necessary requirements of the aerodynamic models and pinpoint weaknesses. Brocklehurst and Barakos [7] give a good overview of today's rotor blade designs focusing on the blade tip shape. The blade tip has a major impact on the blade performance, as most of the lift as well as drag are generated at the outboard stations.

The planform optimization is defined in this work by the specification of twist, taper/chord-length distribution, sweep, and an-/dihedral at specific radial stations. Figure 3.2 depicts a parameterization of them for the later optimized 7A rotor blade. The effect these parameters have on the rotor blade are summarized in Table 3.1, while in the following sections the primary and secondary driving effects are explained.

3.1.1 Twist

This parameter has the strongest effect on the performance of a rotor blade. The twist adjusts the local pitch angle of the blade and thus indirectly the angle of attack, which defines the circulation of the blade. The choice of twist is driven by the inflow, which is induced by the downwash and tip vortices of the previous blades. Through application of the blade element momentum theory (**BEMT**) an ideal twist distribution for hover

Parameter	physics	Advantages	Disadvantages
anhedral increase	vortex field	reduced induced drag, BVI reduction (acoustics)	structural problems
chord decrease	area → inflow distribution, friction	reduced (friction) drag, better circulation distribution	less lift, potential of stall, decreased inertia (autorotation)
sweep increase	compressibility	reduced transonic drag	structural instability
twist (negative) increase	angle of attack → inflow distribution, vortex field	better circulation distribution, reduced induced drag	less lift, potential of stall, vibrations

Table 3.1: Qualitative assessment of the individual parameters, deduced from Prouty [134] and Bittner [135].

is derived by Gessow [136], which is of hyperbolic nature and is mostly dependent on the thrust and airfoil performance. As the tip vortex is neglected in this theory, the real optimal twist distribution looks differently as will be seen in Chapter 4.

The second driver of the twist parameter is the blade torsion. Blade torsion is triggered by the pitching moment arising from non-symmetric airfoils and an offset of the quarter chord line to the elastic axis, which can occur from either blade sweep or anhedral. When using a higher dimensional parameterization for the twist, the optimizer tries to compensate aero-elastic effects by pre-twisting the blade, which is also seen in Imiela [125] and Heilers [137]. If the blade then deforms under the torsional load caused by a swept blade, the optimal twist parameter will change.

According to Bousman [12], twist along with the proper chord length helps to align the airfoils closer to their optimal operational state, but also that a trade-off is to be made between radial sections of the rotor. Eventually it is wiser to generate more lift in a specific region, which also increases drag, if both are reduced in another region yielding an overall reduction of the required power for a fixed thrust.

Therefore, a method for optimizing twist has to be primarily able to capture the downwash and the tip vortices, but for a more accurate design, viscous effects as well as structural coupling effects need to be accounted for. In the presence of shocks, these need to be modelled also, as they change the lift, the drag and the pitching moment of the blade.

3.1.2 Taper/Chord Length Distribution

The chord distribution is driven by similar effects as the twist parameter. Through the increase or decrease of area, the circulation on the blade is modified. An additional driver that affects the blade performance is the friction on the blade. The greater the wetted area, the more viscous drag is generated. Gessow [136] also derives an ideal chord distribution assuming the ideal twist distribution with certain airfoil characteristics, but this theoretical and also hyperbolic chord distribution suffers from the same missing assumptions as the ideal twist derivation namely the tip vortices of the indi-

vidual blades.

The second driver is the resulting sweep of the trailing and leading edge of the blade through tapering. Superimposed with the sweep parameter, very high trailing or leading edge sweep angles are achieved. The aero-elastic effect of the chord distribution is the change of mass and inertia for the specific sections, which also influence the elastic blade deformation. This, however, has little impact on the aerodynamic performance, but more on blade dynamics which are not the central topic of this work.

Thus, methods for the optimization of the chord distribution have similar requirements as for the twist. Inviscid as well viscous and transonic effects are thus necessary to capture all design drivers.

3.1.3 Sweep

The main idea of sweeping the blade is to minimize the transonic drag as the blade tip usually operates in these speed regimes.

Special care is required when this parameter is optimized, because most fluid-structural coupling effects originate from blade sweep. Imiela [125] shows that an optimizer might exploit this parameter to locally optimize the twist of the blade through the resulting torsion instead of only reducing wave drag. Secondary effects are the redistribution of the blade loading as well as spatially delaying or accelerating the formation of tip vortices.

Suitable methods for optimizing this parameter need to be able to simulate compressibility effects. For more detailed designs these methods also need to be able to predict the vortex field around the blade properly as well as viscous effects for correctly representing the aerodynamic torsional loads on the blade. Fluid-structural coupling is mandatory for this parameter.

3.1.4 An-/Dihedral

The an- or dihedral of the blade determines the shedding position of the tip vortex as well as its strength. Decreased vortex strength means less induced power and if the vortex miss distance becomes larger, the interaction of the tip vortex with the next blade is also reduced. Another effect is the influence on the circulation of the rotor, as an an-/dihedral tilts the direction of the resulting aerodynamic force of the particular blade section.

Hollands [9] finds out that a mild dihedral is beneficial in forward flight, while Heilers [137] retrieves a winglet for the rotor in hover. The latter is not applicable to forward flight anymore, as strong aero-elastic coupling effects cause this blade to fail in this flight condition.

For the optimization of the an-/dihedral of a rotor blade, the employed simulation needs to model the tip vortex correctly. Due to the strong influence of the aero-elastic coupling of this parameter, fluid-structural coupling is crucial to take into account. This parameter is also very sensitive with respect to the twist parameter as Heilers demonstrates, thus optimizing it by itself is not recommended.

3.2 Theoretical Review of Aerodynamic Models

In order to model a rotor blade properly, the triple disciplinary task of solving the aerodynamics, structural dynamics and flight mechanics is to be solved. As this work is concerned with the different modeling possibilities of aerodynamics, the portion for solving the structural dynamics and flight mechanics is kept the same, though different fidelities exist for those physics, too. First the various aerodynamic methods analyzed in this work are presented. Then the general blade model along with the coupling process is explained to understand the inclusion of the aerodynamic models in the context of the whole simulation process.

3.2.1 Blade Element Theory

The blade element theory (**BET** [26]) is a simple, effective approach to compute the load distribution over a rotor blade. This is the simplest aerodynamic method available in the here utilized comprehensive code HOST [32]. The blade is discretized by the quarter chord line, where depending on the inflow the aerodynamic forces are computed from airfoil coefficient tables. This yields a robust and quick method to evaluate the performance of a helicopter rotor. However, the major drawback of this method is its limited accuracy. The viscous and compressibility phenomena are only modelled in the two-dimensional format of airfoil tables and the determination of the inflow is a non-trivial undertaking. Therefore in hover the **BET** is extended with the momentum theory, while for forward flight linear inflow models improve the **BET**.

The blade element momentum theory (**BEMT** [136]) , combines the regular blade element theory with a local momentum consideration at the individual element. The local thrust calculated by the **BET** is set equal to the thrust calculated by the momentum theory for an annulus of the same radial location. The resulting equation is iteratively solved for the inflow with which the sectional load is recalculated. The advantage over the regular **BET** is that the inflow representation is better, but the blade vortex interaction is still not captured.

Linear inflow models are usually of semi-empirical nature, which distribute the constant inflow computed by the momentum theory azimuthally for forward flight conditions with a first order harmonics ansatz function. In this study, the Meijer-Drees [138] model determines the inflow in forward flight conditions with the blade element theory. For more details on linear inflow models, see Leishmann [28]

HOST features semi-empirical correction methods, which implement three-dimensional effects in a simplified fashion. The available correction methods for the plain **BET** in HOST are explained by Arnaud et al. [31], which have all been used in this work including those for stall, blade sweep and otherwise curved blades.

3.2.2 Finite State Inflow

The finite state unsteady wake inflow model **FISUW** in HOST is developed by Basset et al. [36]. It takes the idea of Kinner [34], who employs pressure potentials on the rotor disc to solve for the inflow. The equations are derived on the assumption of small incompressible perturbations in contrast to the steady convection airspeed. The resulting partial differential equations are orthogonalized by He [139], who describes the tem-

poral changes by Fourier series and the spatial changes by Legendre polynomials. The resulting pressure potentials allow for the solution of non-linear inflow distributions in the rotor plane.

In HOST, the circulation of the blade is computed with the **BET** and coupled into **FISUW** to solve for the inflow velocities. These are then utilized in the **BET** to correct the aerodynamic loads on the blade. This is coupled back and forth until convergence is reached. Thus **FISUW** enhances the **BET** by non-linear inflow states, however, the blade tip vortex is not taken into account as only the bound vortex potentials are employed.

3.2.3 Wake Models

To overcome the major drawback of the **BET**, the lack of modeling the blade-vortex interaction, wake models are developed. The wake of the rotor is generated by using the Kutta-Joukowski lift theorem and the law of Biot-Savart. First, the bound vortices of the blade are determined by the lift generated of each blade section and then the trailing wakes as well as the vortices are convected. In the last step, the resulting induced velocities on the blade sections are computed with the law of Biot-Savart. This ends up in an iterative procedure as the changed velocities also change the forces on the blade, which again change the vortices.

For the convection of the vortex two major approaches exist. In the first approach, the wake geometry is set based on parameters such as thrust, number of blades and flight condition, referred to as a prescribed wake model (**P.WAKE**). The second approach solves for the wake geometry iteratively by convecting the vortices with the computed induced velocities, referred to as a free wake model (**F.WAKE**). HOST features both methods through the modules METAR [40] and MESIR [45] for the prescribed and the free wake modeling, respectively.

The advantage of wake vortex models is that the induced velocities are resolved more accurately. Blade vortex interactions are captured. However, for the prescribed wake model the influence is geometrically fixed, due to the fact that the wake geometry is pre-determined and does not evolve according to the computed blade geometry. The advantage of the prescribed over the free wake model is the reduction in computational time. While the prescribed wake model only iterates between vortex strength and downwash velocities, the free wake model features an additional loop around this procedure to find the wake geometry. In the derivations of these methods, potential flow is assumed, thus the flow field is incompressible, inviscid and irrotational.

3.2.4 Panel Methods

Panel methods, or surface singularity methods assume the same flow features as the wake models. The surface is described through velocity potentials. The here used panel method is extended with a regular free wake model, where sources/sinks are introduced to describe the surface of the blade along with vorticity potentials. The Kutta-Condition is enforced to close the system of equations, which simply states that a stagnation point exists on the trailing edge panel. This can be implemented in different ways either by ensuring the tangential velocities are zero at the trailing edge, or in a stronger formulation, that the pressure is the same on the top and bottom surface of

the trailing edge. Here the weaker formulation is applied, as it is numerically more stable and can be directly solved.

The employed solver is the Unsteady Panel Method (**UPM**) [140]. The strength of the panel methods over RANS based CFD methods is that they require less computational effort, at the cost of neglecting friction and compressibility effects. Though, the Glauert correction of $1/\sqrt{|1-M^2|}$ is multiplied with the aerodynamic coefficients of each airfoil section to account for the effect of compressibility. This correction becomes invalid in transonic situations, especially when shocks occur.

3.2.5 Computational Fluid Dynamics (CFD)

Computational fluid dynamics refers here to the solution of the Reynolds averaged Navier-Stokes equations and derivatives. The solution of the Euler equations is considered a mid-fidelity method, while the RANS equations are considered as high-fidelity. When computing the Euler solution (**EU**) all inviscid aerodynamic effects of helicopter rotors are included. This includes compressible effects such as shocks, as well as blade vortex interactions, but no flow separation or boundary layer effects. When solving the Reynolds averaged Navier-Stokes equations (**NS**), viscous effects are included and mild separations can also be simulated with a turbulence model. However, typical one- and two-equation turbulence models are not recommended for strong dynamic stall cases.

The major disadvantage of CFD is that it is very resource intensive and the employment of a cluster computer system is necessary. A different problem is that in order to obtain a robust solution and not to violate physics, numerical dissipation or flux limiting is required. This unfortunately has an impact on the conservation of the tip vortices. Special care has to be taken to keep the vortex as long as possible by reducing the numerical viscosity to the absolute necessary minimum, as well as increasing the grid density in regions where the vortices are expected.

The CFD solver in this work is the block structured DLR code FLOWer [141]. The second order central scheme by Jameson [142] is applied in space with a five-step Runge-Kutta scheme in time. The steady solutions are generated using a local-time marching scheme, while unsteady solutions are obtained by a second-order dual-time stepping. Multi-grid and implicit residual smoothing accelerate the convergence. For viscous computations, turbulence is simulated with the two-equation $k-\omega$ SST model by Menter [72]. This turbulence model is able to handle mild-separations, which suffices in representing the here considered design conditions. To be able to handle multiple blades in forward flight, the Chimera [143] technique is implemented into FLOWer.

3.2.6 Flight Mechanical and Structural Blade Model

An accurate aerodynamic simulation requires modelling the blade dynamics and deformation. In all procedures, the comprehensive code HOST [32] computes the blade deformation and the rigid body motion. The way these are modeled is by splitting up the blade into fictive hinges with a variable stiffness in each direction. The general blade dynamics are computed by the Lagrange equations:

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{q}_i} - \frac{\partial T}{\partial q_i} + \frac{\partial U_{el}}{\partial q_i} = Q_i \tag{3.1}$$

with T the kinetic energy, U_{el} the internal elastic energy, and Q_i the external forces caused by the aerodynamic loads. q_i are the generalized coordinates. The kinetic energy is computed from the rigid blade movement and inertia, while the internal elastic energy is obtained from the structural properties and the deformations of the blade. The latter is computed by the classical beam theory, which is considered valid for long slender wings, which rotor blades are. As the direct solution results in a large linear system, a modal ansatz exploits the periodicity of the problem.

During the numerical optimization it is essential to compare the rotors at the same operational state, e.g. the same thrust, roll- and pitch moments. Therefore it is mandatory to trim the currently evaluated design. A Newton-Raphson iterative method achieves this, which seeks the state of equilibrium for a set of forces and moments in dependence of the input control angles and rotor attitude. The trim objective, the forces and moments which are trimmed, as well as the input quantities, will be stated with the considered flight condition.

3.2.7 Coupled Simulations

The prediction of the rotor performance requires to couple aerodynamic, structural and dynamical models together. To understand the full process to accomplish this task, the here utilized procedure is illustrated in Figure 3.3, an extension of the figure in [40]. First the initial starting point is selected, which includes the settings for the initial elastic deformation, control angles and rigid body motion. Then, one of the above explained aerodynamic methods computes the external forces required to determine the elastic deformations. With the external forces and deformations found, the rigid body motion, and thus the trim procedure is executed. This procedure is repeated until equilibrium is found.

For CFD solutions, first a converged trim solution is generated with the **BET** in HOST. Then a weak coupling approach is applied to include the 3D forces computed with FLOWer for the given blade deformation and motion. Then the trim procedure is repeated in HOST, with the aerodynamic sectional forces now determined by the following equation:

$$\vec{F}^n = \vec{F}^n_{2D} + \Delta\vec{F}^{n-1} = \vec{F}^n_{2D} + (\vec{F}^{n-1}_{3D} - \vec{F}^{n-1}_{2D}) \tag{3.2}$$

with $\vec{F}^n$ being the current aerodynamic forces, and $\vec{F}^{n-1}$ the aerodynamic forces from the previous coupling step. The term $\Delta\vec{F}^{n-1}$ of the last step are the delta loads obtained from the difference of the **BET** ($\vec{F}^{n-1}_{2D}$) and CFD ($\vec{F}^{n-1}_{3D}$) from the previous coupling step. The delta loads correct the prediction of the **BET**, which are fixed during the trim procedure and only the current 2D forces $\vec{F}^n_{2D}$ are altered. With a new trim solution available, the 3D forces are recomputed with the new deformations and trim settings. This is repeated until the change in deformations and control angles become sufficiently small. Then the 2D loads from the current and last coupling step are equal and thus the aerodynamic loads in the trim procedure purely consist of the 3D forces. A more detailed discussion on the HOST-FLOWer coupling is found in Dietz et al. [74].

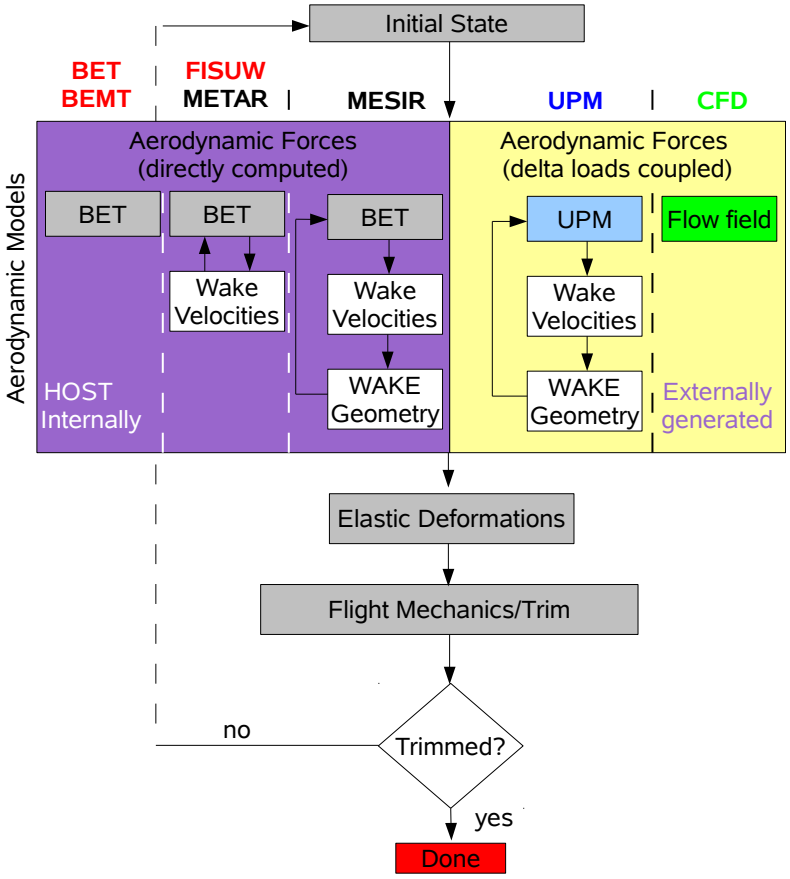


Figure 3.3: General rotor simulation procedure

In this work the coupling process is stopped, if the change in required rotor power from one to the next coupling step is less than 0.1%, which usually is associated with changes in the control angles of less than $0.1°$.

At the time of preparing this thesis, **UPM** is not able to account for the aero-elastic deformations. Therefore, only the initial trim condition is determined by HOST and the final trim is obtained by the internal trim procedure within **UPM**. The methodology is also based on a Newton method.

3.2.8 Theoretical Fidelity of the Different Aerodynamic Models

Looking at the different strengths of the methods, the models are categorized into three levels; low-, mid- and high-fidelity simulations. The border cannot be sharply distinguished, but the author places all the blade element based approaches, even with the enhancement of wake models, into the category of low-fidelity simulations. Reason for this is that real 3D effects are still difficult to impose onto 2D airfoil sections, despite the fact that the wake models allow to compute the induced velocities acting upon them. The mid-fidelity section is comprised of the inviscid methods, namely the panel method enhanced with the wake model, as well as Euler CFD computations. In the panel method, the blade is treated as a surface and the effects of the different induced inflow velocities are computed on the exact location of the blade within the discretization accuracy. It must be highlighted, however, that the level of fidelity can-

not necessarily be raised for all physical effects when going from the proposed low- to mid-fidelity methods. The field data might be more correct with the here defined mid-fidelity methods, yet friction is completely neglected, which is not the case for the BET methods. As the high-fidelity method, RANS simulations pose the minimum to capture all effects, though laminar-turbulent transition prediction is neglected here and highly separated flows will also be difficult to properly predict. The latter is assumed to be of no major impact, as the here considered design points do not lie in regions of high flow separation, such as a maximum thrust or dynamic stall flight conditions. An overview of the different physical aspects of each method is presented in Section A.2.1, where a comparison of all introduced methods is tabled along with the physical effects they are able to model. Table 3.2 gives an overview of how the different parameters mentioned in Section 3.1 are pictured in these methods.

Model/Parameter	anhedral	chord	sweep	twist
BET/ **BEMT**	-	tabled data	corr. factor	inflow models/ momentum theory
FISUW	-	tabled data	corr. factor	finite state potentials
Wake models (**P.WAKE**/**F.WAKE**)	wake model	wake model tabled data	corr. factor	wake model tabled data
UPM (**UPM**)	wake model	potential flow	corr. factor	potential flow, wake model
CFD Euler (**EU**)	flow field	potential flow	flow field	flow field
CFD RANS (**(F)NS**)	flow field	viscous flow	flow field	flow field

Table 3.2: Representation of planform parameters by different aerodynamic models

3.3 Application of Aerodynamic Models to Different Rotor Geometries

After the theoretical discussion of the aerodynamic models in Section 3.2, as well as the planform parameters and their physical effect on the rotor blade in Section 3.1, this section covers the investigation of the different methods as to how well they reflect the variation of rotor parameters. As already found out in Chapter 2, the low-fidelity has to correctly resemble the high-fidelity model, which has to be examined first for the given aerodynamic models.

For this study, the 7A rotor blade [144] is chosen as the initial reference configuration as LePape [122] and Imiela [125] also optimize this blade. They explore the behavior of the different planform parameters, anhedral, chord, sweep and twist in two different flight conditions which are compared with the results in this work.

The parameterization and flight conditions chosen for the 7A rotor are explained in Section 3.3.1. The baseline 7A rotor is then simulated with each method and the differences among them are highlighted in Section 3.3.2. In Section 3.3.3 a sensitivity study of the aforementioned design parameters is undertaken in order to obtain a sense of the behavior of the different blades with respect to the performance, here the required

power of the rotor. The investigation is finished in Section 3.3.4 with a direct genetic optimization, for which it is assumed that the global optimum is found for each method. From this, a selection of models suitable for blade optimization at low-, mid- and high-fidelity is deduced in Section 3.4, also considering the fitness for variable-fidelity optimizations.

3.3.1 Parameterization and Flight Conditions

The 7A rotor is a model rotor with $2.1m$ radius. It has a rectangular planform, an aerodynamic linear twist and two airfoils. The OA213 airfoil has 13% thickness and is located at the root section, while the OA209 airfoil has 9% thickness and is located at the blade tip. At $1.575m$ radius, the airfoil transitions linearly from the OA213 to the OA209 until a radius of $1.89m$. In the here chosen test cases it is operated with a tip Mach number of 0.65. In both flight conditions, hover and forward flight, the vertical force is trimmed to $4400N$, while in forward flight a virtual fuselage drag of $530N$ has to be overcome at a flight Mach number of 0.25 corresponding to an advance ratio of $\mu = 0.38$. The parameterization is based upon the one given by Imiela [125]. The application of the design parameters is depicted in Figure 3.2 and the parameter boundaries are listed in Table 3.3 for both forward flight and hover. The parameters anhedral, chord and sweep are varied with quadratic function starting at the blade tip, while the aerodynamic twist is varied linearly over the blade span. The blade tip starting position is set to 80.6 % rotor radius $\approx 1.7m$.

Parameter	baseline value	lower bound hover/forward flight	upper bound hover/forward flight
anh. [$\cdot c_{ref}$]	0.0	-1.0/-0.3	1.0/0.3
chord [$\cdot c_{ref}$]	1.0	0.5	1.5
sweep [$\cdot c_{ref}$]	0.0	-1.0/0.0	1.0
twist [°]	-4.32	-20.0/-16.0	0.0/-4.0

Table 3.3: Parameter boundaries of 7A optimization. Reference chord length $c_{ref} = 0.14m$. hv=hover and ff=forward flight variables

Anhedral and sweep are achieved by offsetting the blade tip by a specific value based upon the reference chord length, while the blade taper is applied through decreasing the blade tip chord length. When the latter is modified, the root chord length is adjusted by ensuring a constant thrust weighted average chord length equal to the 7A chord length of $0.14m$. The internal structural properties, such as inertia and blade stiffness, remain unmodified and only the aerodynamic moments caused from anhedral and sweep are included in the elastic model. This is considered valid in this initial study due to the rather small changes made and the focus is laid on the aerodynamic models.

3.3.2 Baseline Rotor Results

The 7A rotor blade is modeled with each aerodynamic method listed in Section 3.2 with the naming convention provided in Table 3.4. The numerical settings and individual discretization of each method are listed in Section A.2.2. Hence the **EU** and

Tag	Description
BET	plain blade element theory with all correction factors in HOST.
BEMT	blade element momentum theory, only available for hover computations.
FISUW	the BET enhanced with the finite state inflow model FISUW in HOST.
P.WAKE	the BET enhanced with the **prescribed** wake model METAR in HOST.
F.WAKE	the BET enhanced with the **free** wake model MESIR in HOST.
UPM	panel method including an unsteady free wake model, the selected code is UPM.
EU	use of the inviscid Euler equations in FLOWer. Periodic meshes are used in hover and single blade simulations are used in forward flight.
NS	use of the viscous RANS equations in FLOWer on a coarse grid. Periodic meshes are used in hover and single blade simulations are used in forward flight.
FNS	use of the viscous RANS equations in FLOWer on a fine grid. Periodic meshes are used in hover and a multi-blade setup is used in forward flight.

Table 3.4: Naming convention of the investigated aerodynamic models

NS only model a single blade grid in forward flight, this implies that the effect of the tip vortex of the previous blade is left out of consideration. Imiela [125] produces almost matching results during the optimization in comparison with full-bladed meshes and in Imiela and Wilke [3] it is observed that for advance ratios greater $\mu > 0.2$ the tendencies of the speed-power curve are very similar. Yet, no direct comparison of optimizations with one or multiple blades has been made. It is assumed that the highest attainable fidelity through aerodynamic modeling is achieved with the Navier-Stokes solution on the fine meshes (**FNS**) and is considered the reference.

The resulting power required for each rotor is plotted in Figure 3.4. The results of the momentum theory (MT) along with experimental data are also presented in this chart. The momentum theory allows computing the minimal required power of the rotor blade by assuming incompressibility and uniform inflow without any viscous losses. The required power for hover is extracted from Beaumier et al. [145] and for forward flight from Pahlke and van der Wall [146] for the experimental references. The blue bars denote the hover results and the red bars the forward flight results.

In hover, the **BEMT** method yields the best result followed by the **FNS** method. This shows that the **BEMT** already generates good results, but CFD also re-produces the experimental results well. Comparing the **EU** and **FNS** method, a relative difference of about 20% is recorded. This difference is attributed to the lack of friction for the **EU** method and looking at the output of **FNS**, this difference is also reported by the difference of pressure and viscous drag by the flow solver. The wake methods **P.WAKE** and **F.WAKE** return very low values for the required power. This is also discovered by Kocurek and Tangler [41], who are 15% too optimistic in their prediction with a lifting-line model. They say that lifting line models tend to overreact to the tip vortex in hover and claim that lifting surface methods improve on this fact. However, when comparing with the **UPM** results, this is rather questionable. The required power obtained from **UPM** is $56.15\ kW$, which is very close to the $52.59\ kW$ predicted by the

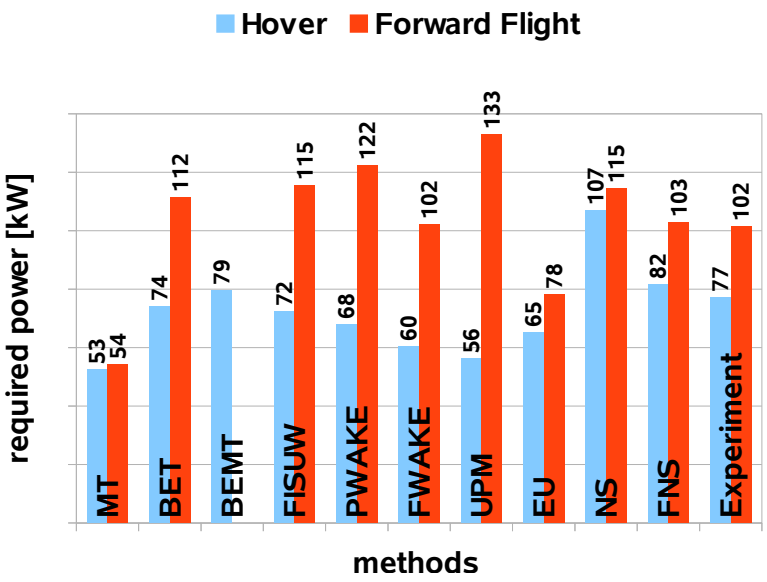


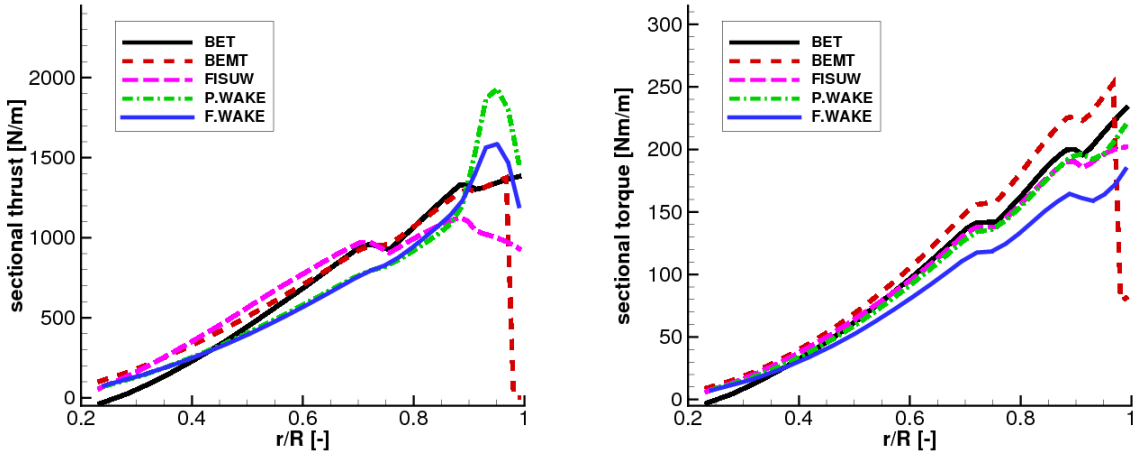
Figure 3.4: Bar chart of the required power computed by each method for hover and forward flight.

momentum theory. The **NS** method largely over predicts the required power, which is reasoned with the generally higher dissipation in this method coming from the coarse grid resolution and the need to add numerical viscosity to guarantee robustness of this process. This is lessened for the **EU** simulation as no physical viscosity exists. Since the computed viscous power required to operate the 7A rotor is very similar between the **NS** and the **FNS** method, the overall resolution of the boundary layer is sufficient for both methods.

The required power in forward flight is greater than in hover, a trend that all methods also predict. Here the **F.WAKE** is closest to the experimental results and the **FNS** is almost on par with it. The **EU** differs by 23.9% from the **FNS** solution. In contrast, the **FNS** solution has on average a difference of 20% of the pressure to the viscous drag. This indicates that the **EU** solution already captures most inviscid effects and only the viscous effects are missing. The greatest outliers are the **UPM** and the **P.WAKE** method in this flight condition, which largely overestimate the required power.

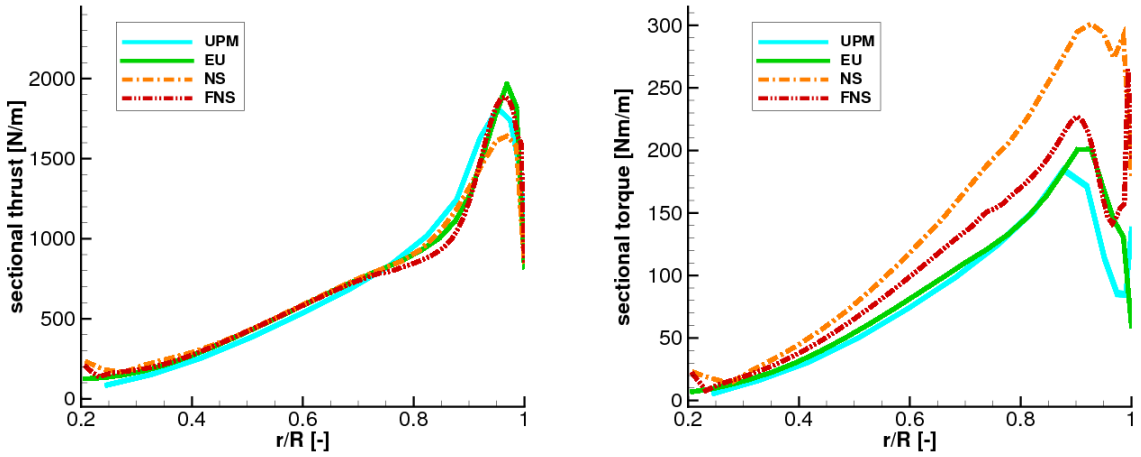
While the absolute power is good for a first comparison of the methods, the more important part for the suitability of optimizations is a valid prediction of the minima for varying design parameters. This is only reached, if the physical representation is similar. It can happen that the absolute value aligns very well, yet only because of cancellation of errors. For example tip losses are ignored, but the friction drag is over predicted for the examined configuration.

To visualize the impact of the different aerodynamic models onto the rotor performance, the sectional loads for hover are plotted. Figure 3.5(a) and Figure 3.5(b) plot the thrust and torque for the blade element theory based methods and Figure 3.6(a) and Figure 3.6(b) for the **UPM** and the CFD simulations. The sectional thrust plots help to find the vortex passage from the previous blade characterized by a kink around 80% rotor radius followed by a peak in the thrust distribution. Thus, the thrust plots indicate regions of interest for the parameterization. The torque plot is good for spotting points where changes are necessary to reduce the overall required power and are more suitable than the drag distribution as the radial weighting is included.



(a) Thrust (b) Torque

Figure 3.5: Comparison of sectional thrust and drag among the low-fidelity line methods for the baseline blade in hover case.



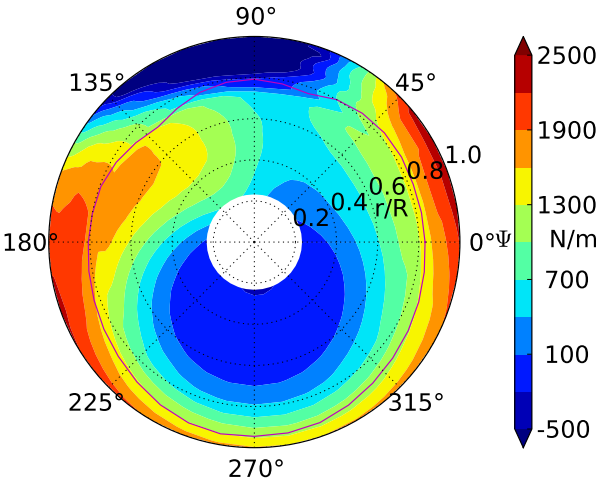
(a) Thrust (b) Torque

Figure 3.6: Comparison of sectional thrust and drag among the surface and volume methods for the baseline blades in hover case.

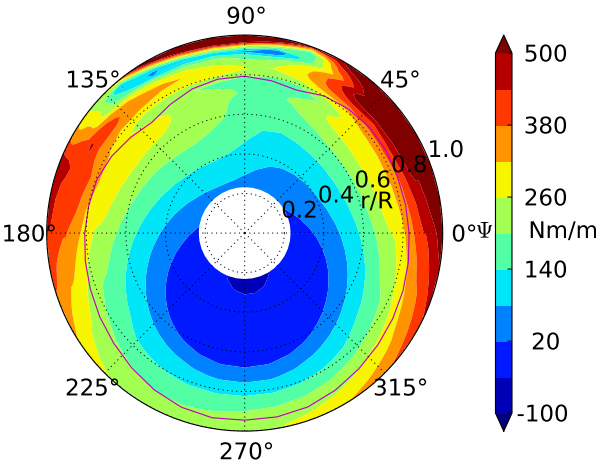
Looking at the **BET**, **BEMT** and **FISUW** method, the peak in the thrust distribution is not found, which clearly is coming from the lack of blade tip vortex modeling. The kink at about $0.75\ r/R$ of the blade marks the transition from the thick OA213 to the thin OA209 airfoil, which are aligned with their zero lift angle attack for the 7A blade. This effect is greater in the torque distribution. Only the **BEMT** includes a tip-loss factor in HOST, while the wake models have natural tendencies to implement such a loss in the thrust distribution. Though, in the torque distribution, most methods are quite similar in the trend they predict and only the slope of the curve changes.

Going over the mid and high-fidelity methods, the impact of the vortex passage is better taken into account as the tip loss is also spotted in the torque plot. The **P.WAKE** and **F.WAKE** method resolve the tip loss coming from the tip vortex in the thrust distribution, but this tip loss is not seen in the torque distribution. The results are similar to the **FISUW** solution. **UPM** and the CFD methods resolve the kink in both, the thrust and torque distribution. The kink in the thrust distribution is higher for the **EU** method than for the **FNS** method. The torque distribution of the **EU** is similar to the **UPM** distribution, which are both lower than the one from **FNS**. The latter is attributed to the lack of friction in both methods. The thrust distribution of the **NS** method is more gradual and the peak is less than with the **FNS** method. Along with the noticeable increase of torque, this is argued with the added numerical viscosity of the coarser grid for the **NS** method over the **FNS** method.

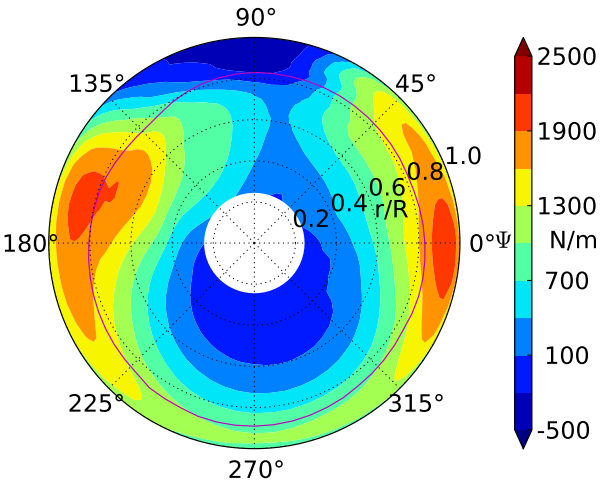
Comparing forward flight capabilities, Figure 3.7(a) to Figure 3.7(h) depict the rotor disc plots for four selected methods. On the left side of Figure 3.7, the thrust distribution is plotted, on the right side the torque distribution. The **BET** predicts a strong downforce on the advancing side around $\psi \approx 110^o$, which lessens when increasing fidelity. For the torque distribution, an artefact is found for the **BET** on the advancing side, which is also visible for the **P.WAKE** method. This artefact does not exist for the **EU** and **FNS** method, but also for the not shown **FISUW** and **F.WAKE** solutions, which leads to the conclusion that this arises from false two dimensional aerodynamics. **UPM** resolved a very different thrust and torque distribution. Due to the high advance ratio, almost 40% of the retreating blade are in reverse flow, which makes it difficult to ensure the Kutta condition at the trailing edge. Therefore, only partial convergence can be achieved and the result is highly erroneous. The assumption that the tip-vortices are convected out of the rotor system quickly enough to neglect them in the simulation is validated by the single-blade approach of **EU** for this high-advance ratio flight condition. Both, thrust- and torque distribution of the **EU** method resemble the **FNS** solution closely. The torque distribution of the **EU** method is lower almost on the whole disc, which arises from the lack of friction in this simulation. As expected, the not plotted **NS** solution improves on this fact and resembles the **FNS** solution better than the **EU** method due to the included viscosity.



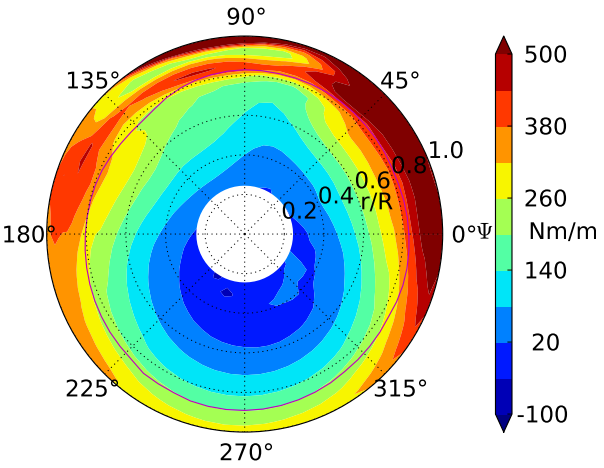
(a) Sectional thrust for **BET** method



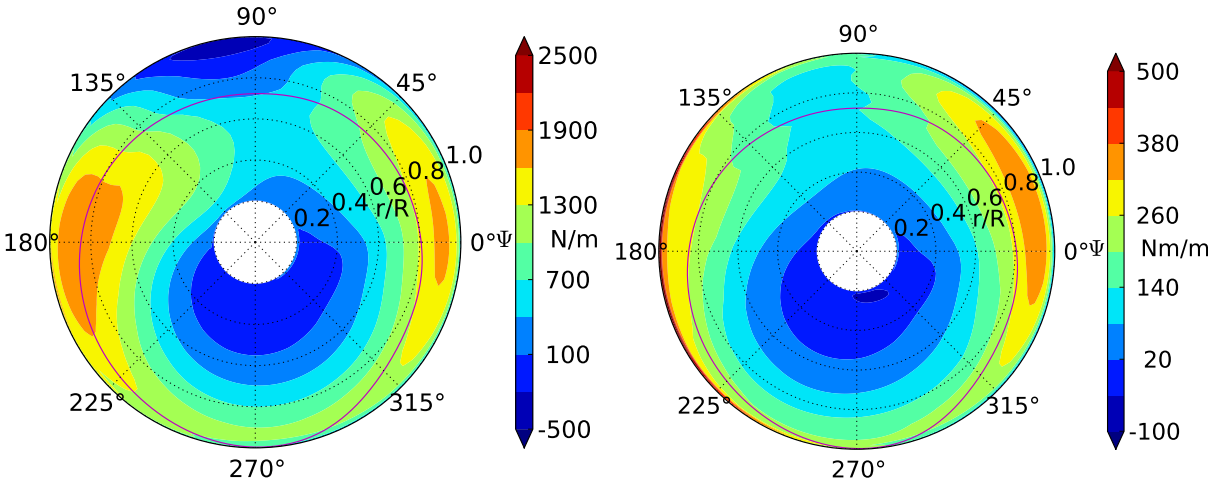
(b) Sectional torque for **BET** method



(c) Sectional thrust for **P.WAKE** method

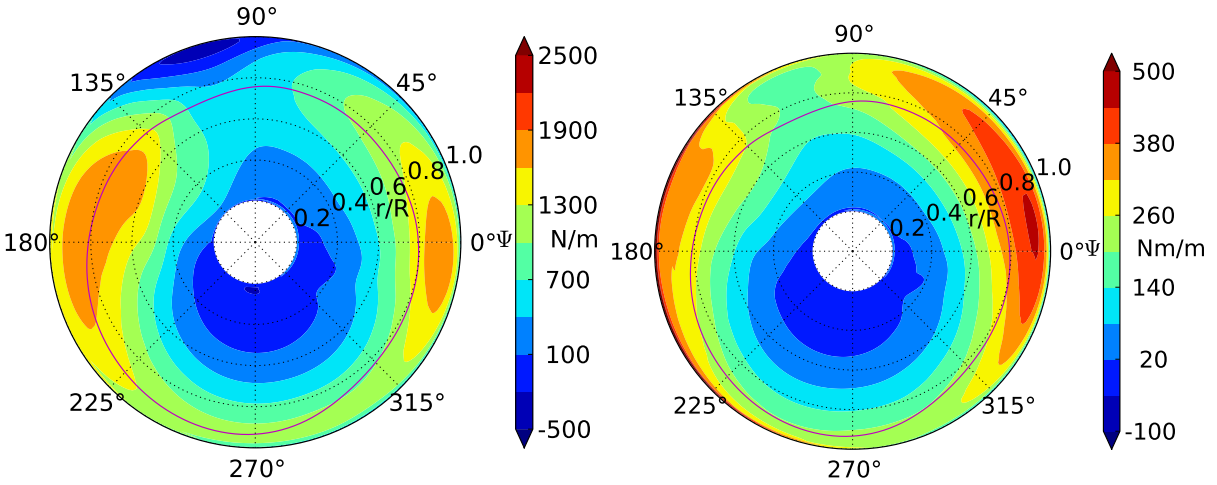


(d) Sectional torque for **P.WAKE** method



(e) Sectional thrust for **EU** method

(f) Sectional torque for **EU** method



(g) Sectional thrust for **FNS** method

(h) Sectional torque for **FNS** method

Figure 3.7: Comparison of sectional thrust and torque among for various methods for the 7A blade in forward flight.

3.3.3 Sensitivity Study

The goal of the sensitivity study is to grasp the behavior of the goal function with respect to geometric changes. Figure 3.9(a) to Figure 3.12(b) graph the parameter variations in hover and forward flight with the legend given in Figure 3.8. The plots show the required power normed as a power fraction, which is defined as the ratio of required power of the evaluated design in contrast to the required power of the reference rotor when computed with the individual method. Thus, the reference rotor always has a power ratio of 1.0 independent of the method. The purpose of this normalization is to emphasize the effect of a parameter change over the single absolute value. From these plots it is observed that groups of similar methods, BET and inflow enhanced BET (**BET**, **BEMT**, **FISUW**) or BET with wake models (**P.WAKE**, **F.WAKE**), the panel method (**UPM**), or the CFD methods (**EU**, **NS**, **FNS**), yield tendencies alike of parameter preference stemming from the similarity in their physical assumptions.

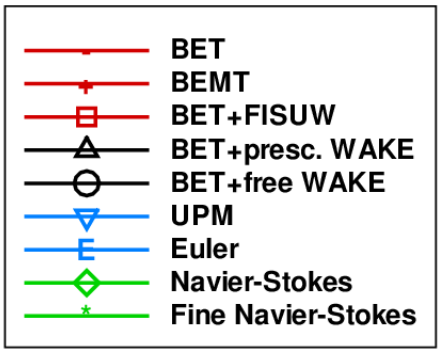
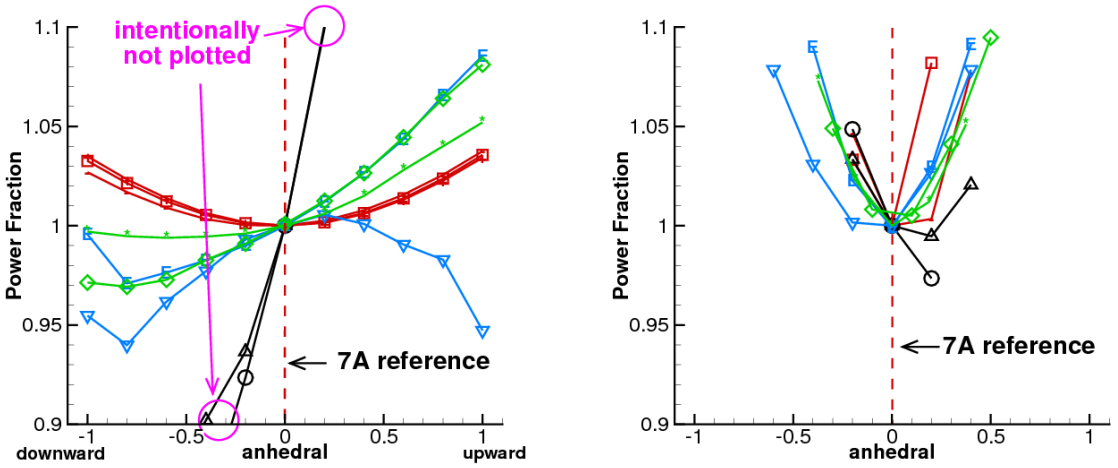


Figure 3.8: Legend for the sensitivity figures

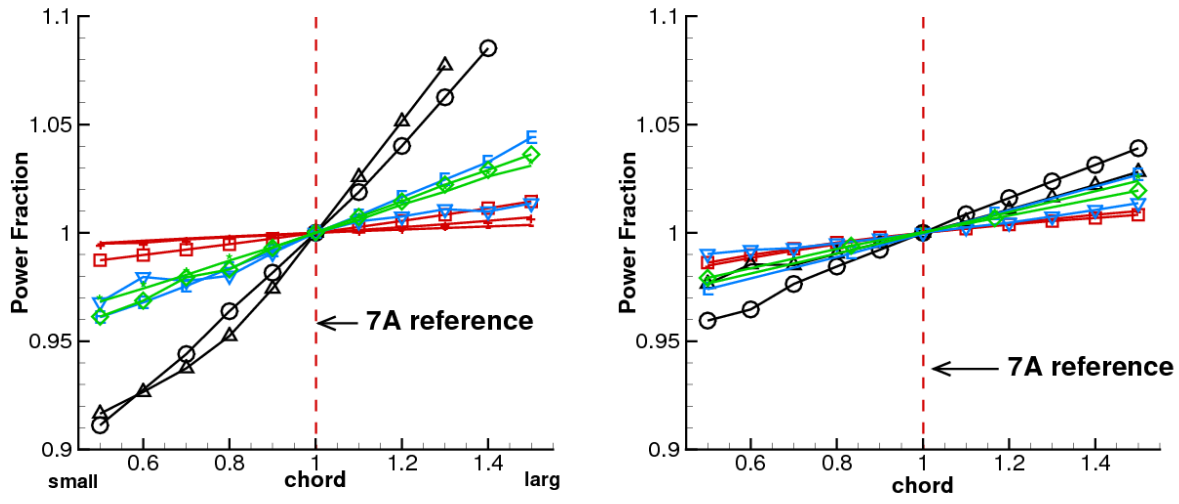
In hover, the results are widespread. The anhedral parameter (Figure 3.9(a)) only has an effect on the structure through the additional lever of the aerodynamic forces when using low-fidelity methods (**BET**, **BEMT**, **FISUW**). The wake coupled methods capture a different trend as the CFD methods. The optimum of the anhedral parameter for the wake models lies at -1 with a power reduction to 82% and 75% for **P.WAKE** and **F.WAKE** respectively, but is intentionally not plotted in Figure 3.9(a). These results are below the induced required power predicted by the momentum theory, which should physically not be possible. **UPM** predicts a mirrored behavior to the low-fidelity methods, which on the downward facing side is similar to the CFD methods, but on the upward side is not captured by any other model. The **NS** and **EU** methods are aligned and have a similar behavior as the **FNS** method, which however predicts a smaller overall effect of this parameter. The benefits of an anhedral or dihedral in hover are greater than in forward flight (Figure 3.9(b)), where the slight aerodynamic gain is lost due to the aero-elastic effects. Points beyond an absolute value of ± 0.3 anhedral are not trimmable for certain methods and thus are not part of Figure 3.9(b). Almost all methods agree here, except the **F.WAKE** method predicting an improvement for increasing dihedral.

The trend of the tip chord parameter, Figure 3.10(a) for hover, Figure 3.10(b) for forward flight, which tapers the blade, is captured similarly for all methods and flight conditions. As a smaller chord length off-loads the blade on the outboard tip, the induced velocity and drag are reduced. The viscous drag, if modeled, has the greatest lever with respect to torque at the blade tip. An almost linear tendency is found in



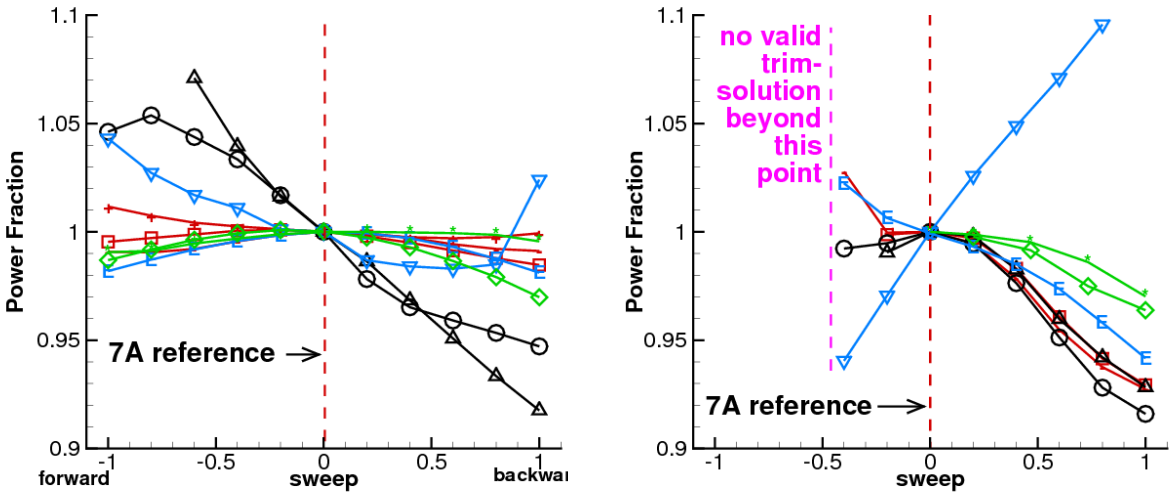
(a) Hover (b) Forward flight

Figure 3.9: Anhedral variation, legend shown in Figure 3.8



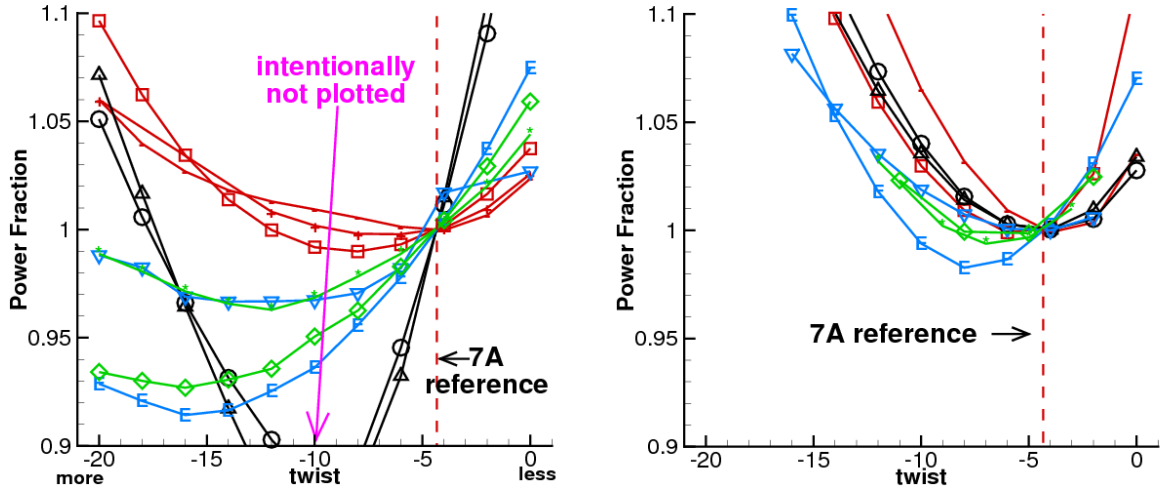
(a) Hover (b) Forward flight

Figure 3.10: Chord variation, legend shown in Figure 3.8



(a) Hover (b) Forward flight

Figure 3.11: Sweep variation, legend shown in Figure 3.8



(a) Hover (b) Forward flight

Figure 3.12: Twist variation, legend shown in Figure 3.8

both flight conditions. In forward flight the various methods have a better overall agreement as the slopes are closer together. Still, the simple wake enhanced methods **P.WAKE** and **F.WAKE** over predict improvement and degradation. This effect is very strong in particular in hover.

Blade sweep, Figure 3.11(a) and Figure 3.11(b), brings an improvement of the goal function for all methods and flight conditions. In hover flight, two optima exist for **BET**, **FISUW** and the CFD methods; a forward swept blade is slightly more favorable than a backward swept blade for **EU** and **FNS**. For all other methods, backward sweep is perceived more valuable with the **P.WAKE** and **F.WAKE** having an exaggerated perception. In forward flight, due to strong blade torsion at the tip caused by the blade offset, only backward sweep is advantageous. In fact, certain values cannot be computed as trimming the blade proves to be impossible. The **UPM** method has an infeasible preference of forward sweep in forward flight as a result of missing elastic modeling.

The power fraction as a function of the twist resembles a parabola in both flight conditions for all methods, Figure 3.12(a) and Figure 3.12(b). Yet, the position of the minimum varies across the methods and flight conditions. The low-fidelity methods usually have the optimum at low twist angles, but with increasing fidelity this value rises. This is attributed to a better downwash prediction. Again, the **P.WAKE** and **F.WAKE** reflect the twist change very differently in hover, while **UPM** and **FNS** resolve almost the same optimal location. They both model the tip vortex equally well. The coarser **EU** and **NS** methods resolve a more diffused, weaker vortex, thus tend to larger twist angles. The simple BET based methods have little preference for large twist angles and almost prefer the setting from the 7A rotor. In forward flight, the various methods have, as with all parameters so far, a better agreement than in hover. The CFD methods tend to slightly bigger twist angles in contrast to the non-CFD methods.

Overall, it is noticed that the wake coupled methods emphasize the effect of parameter variations stronger in the hover flight condition than the other methods. For example the effect on the anhedral parameter in hover grants a reduction of required power of over 19% for **P.WAKE** and over 21% for **F.WAKE**. For **F.WAKE** this leads to an absolute value for the required power of $45.3kW$ in hover, which is below the theoretical value of $54.1kW$. Results from these methods should thus be looked upon with skepticism and are reasons to exclude these models for hover optimizations.

Concluding from this sensitivity study, the downwash is the main driver for the hover case deduced from the strong effect of blade twist. In forward flight, the load redistribution towards the inboard stations achieved by the blade sweeping and tapering brings the most improvement. The an-/dihedral has a larger effect in hover depending on the correct accounting of the tip vortex in the simulation.

3.3.4 Direct Optimization

For the purpose of identifying suitable aerodynamic models for a variable-fidelity rotor optimization, a direct optimization using the genetic algorithm SOGA [147] in the Dakota toolbox [148] is performed with each method and each flight case. The criteria for evaluating the various aerodynamic methods are their robustness, physical feasible modeling of the required power and the computational time required to execute the simulation. The advantage of using a genetic algorithm is that the likelihood of finding

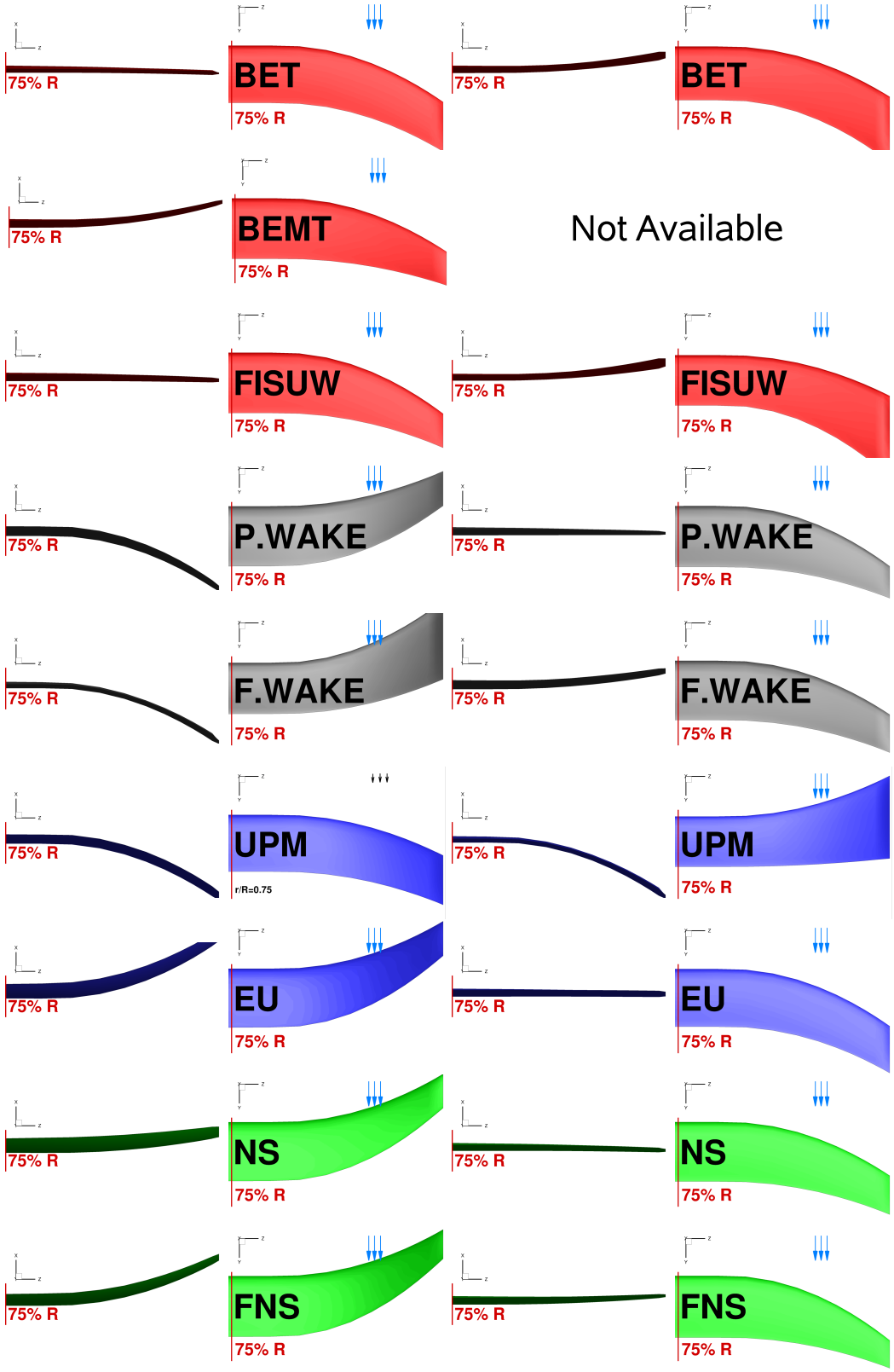


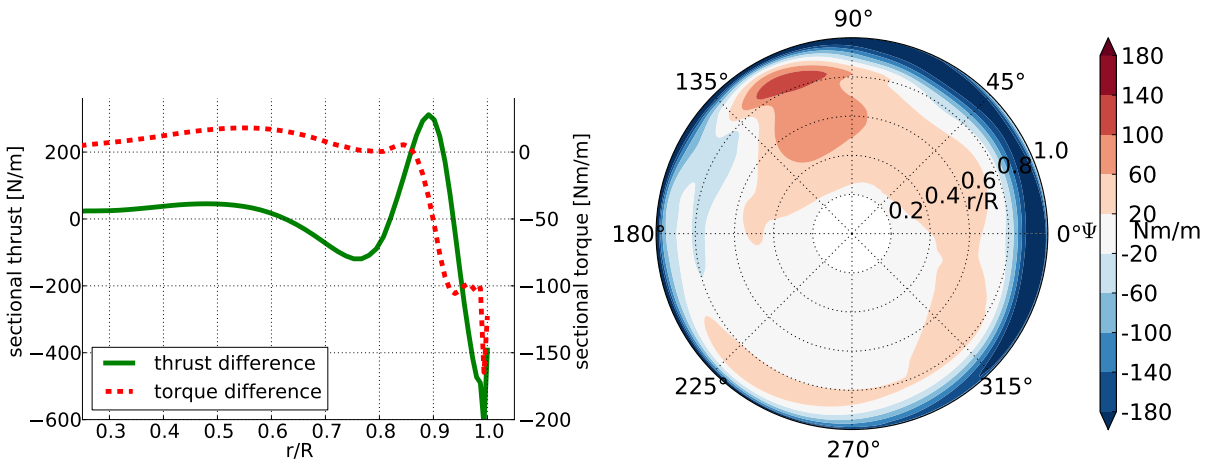
Figure 3.13: Tips of best blades from the optimizations with the different methods.

Method	Power fraction P/P_{ref}	anh.	chord	sweep	twist	Validation with FNS $P_{FNS}/P_{FNS,ref}$
BET	0.986	-0.1	0.9	1.0	-1.7	1.009
BEMT	0.988	0.4	0.6	0.8	-6.0	0.981
FISUW	0.979	-0.0	0.6	1.0	-5.5	0.968
P.WAKE	0.749	-1.0	0.6	-0.7	-9.2	0.965
F.WAKE	0.337	-1.0	1.4	-1.0	-2.6	1.050
UPM	0.767	-1.0	0.9	0.7	-9.4	0.985
EU	0.879	1.0	0.6	-1.0	-16.6	0.938
NS	0.898	0.3	0.6	-1.0	-16.6	0.944
FNS	0.929	0.8	0.5	-1.0	-12.3	0.929

Table 3.5: Final results of different 7A optimizations in hover

Method	Power fraction P/P_{ref}	anh.	chord	sweep	twist	Validation with FNS $P_{FNS}/P_{FNS,ref}$
BET	0.926	0.3	1.2	1.0	-5.5	0.989
FISUW	0.866	0.3	1.5	0.9	-5.5	1.009
P.WAKE	0.928	-0.0	0.6	0.9	-4.5	0.952
F.WAKE	0.868	0.3	0.7	0.9	-7.7	0.957
UPM	0.577	-1.0	1.5	-0.6	-2.6	1.380
EU	0.904	-0.0	0.8	1.0	-5.5	0.955
NS	0.985	-0.0	0.7	0.9	-4.1	0.964
FNS	0.941	0.1	0.5	1.0	-5.3	0.941

Table 3.6: Final results of different 7A optimizations in forward flight



(a) Best hover minus baseline rotor

(b) Best forward flight minus baseline rotor (torque)

Figure 3.14: Comparison of sectional loads in hover and forward flight between the best rotor and the base line configuration for the FNS method.

a globally valid optimum is high. A major drawback of the genetic optimization is that it is very resource consuming and the local accuracy is limited. An exception is made for the four-bladed Chimera computation in forward flight using RANS, for which the optimum is found through the single-fidelity optimization methodology described in Chapter 2, as direct optimization is too costly. The results of the direct optimization are weighted the most in the process of selection, as contradictions in the outcoming designs are mostly traced back to misleading modeling of physical effects. Yet, a configuration that resembles the high-fidelity **FNS** configuration does still not guarantee valid physics.

Hover

In Table 3.5, the metrics for the best rotor of each hover optimization is listed. The column featuring **FNS** contains the power fraction of these individual rotor configurations, when they are re-computed with **FNS**. The idea is to see how much the optima correspond with each other. The visualizations of the different configurations are pictured in Figure 3.13 on the left side. The low-fidelity methods do not utilize the anhedral parameter. The wake coupled methods all do prefer an anhedral, while the CFD methods tend to a dihedral. **NS** is not driven to such a high value of dihedral, given the fact that the tip vortex dissipates faster on the coarse mesh. The **EU** method has a similar trend as **FNS**, as the overall dissipation, numerical together with physical, is less than for **NS**. All methods except **F.WAKE** yield a tapered tip, which corresponds to the findings of the sensitivity study. The varying preference of the sweep parameter of either backward or forward sweeping of the blade depends on the method, just as noticed in the sensitivity study. From investigating single individuals of the **FNS** optimization, it is found that the required power is a multi-modal function originating from forward and backward sweeping. For the twist parameter, a similar result to the sensitivity study is found; with growing fidelity the tip twist angle grows larger, except for **F.WAKE**. The final angle is larger for **EU** and **NS** than for **FNS**.

The methods **BEMT**, **FISUW**, **P.WAKE**, **UPM**, **EU** and **NS** delivered an improvement of the base line rotor in terms of power consumption after recomputing the optima with **FNS**, yet the outcomes are very different. From a designer stand point of view, the **FNS** blade and similar blades are looked upon critically. The forward sweep along with the large dihedral are not suited for forward flight. A solution for this configuration is not obtainable for the forward flight condition. This rotor embodies the optimal solution for this set of parameters and this flight condition for a good aerodynamic design without taking into account any constraints, nor multi-disciplinary considerations. Comparing the sectional loading of the 7A rotor and the optimal **FNS** configuration in Figure 3.14(a), it is seen that the optimized blade generates a lot more thrust on the outboard stations than the reference rotor, but draws a lot less power further out. Reason for this is that the tip vortex induces an upwash past 90% r/R and thus the optimizer tries to place the blade in such a way that this is exploited. The ideal rotor, as derived from the blade element theory in [28], is not attainable. Reason for this is that the **FNS** method does not neglect the tip vortex, nor is the inflow determined upon the incompressible momentum theory, such as in the derivation. The torque loading leads to the conclusion that the optimizer trades off the inboard stations for the outboard stations. Due to the highly twisted blade, the inboard stations perform less efficient, while the outboard stations are better positioned. A non-linear twist would be an alleviation to this.

Forward Flight

In a similar fashion to the hover case, Table 3.6 summarizes the findings from the forward flight optimization with the blades depicted in Figure 3.13 on the right side. In forward flight, the results are more in harmony than in hover. **BET**, **FISUW** and **UPM** are misleading in terms of the chord parameter, for which the optimization process features a widening of the blade towards the tip. Furthermore, **UPM** has an optimum analogue to the hover optimum characterized by the preference for anhedral and forward sweep. The lack of structural and dynamical modeling is responsible for this. The other solutions indicate no strong favoring of dihedral, a full backward sweep, and a tip twist slightly larger than the baseline. The final values of the design parameters are very similar to the ones from the sensitivity study, which leads to the conclusion that the parameters are less aerodynamically coupled in forward flight than in hover. In this case, the **P.WAKE** and **EU** method are well aligned to the **FNS** optimum with **P.WAKE** being slightly better.

Again, comparing the baseline rotor with the optimal **FNS** configuration, Figure 3.14(b) plots the difference in torque between these two. Blue marks the area where the new rotor draws less power, and the red area indicates the opposite; it draws more power. In reference to Imiela's results [125], the improvements noted on the rotor disc are like a superposition of the beneficial effects of tapering the tip, sweeping and re-twisting the blade along with the slight anhedral. The taper decreases the torque beyond 85 % r/R over the whole azimuth, while the twist does this in the range of $\psi = 0° - 150°$. The sweep operates almost in the same manner, which is more or less coming from the aero-elastic tailoring effect. The mild dihedral causes a reduction in the area of $\psi = 120° - 240°$ at the middle to outer radial stations r/R = 0.5-1.0. Though, it is not possible to directly trace back what parameter contributed the most, as the overall improvement is not the sum of improvements from the individual parameters obtained from the sensitivity study. A vague guess is made as to attribute most of the changes to the chord and sweep parameters.

3.4 Choice of Fidelities

From the theoretical discussion in Section 3.2 and the applied investigation in Section 3.3 of the available aerodynamic models, the choice of viable fidelities for the aerodynamic optimization of helicopter rotor blades is made in this section. The outcome of the previous sections is that the correct inflow modeling is essential with the tip vortex being a dominant driver in hover. As the associated effects are mostly pressure based, the effect of viscosity onto the rotor performance for varying planforms is small.

CFD methods are almost exclusively necessary for hover optimizations, as they are the only methods able to capture the inflow correctly. However, as the run times for even the low-fidelity CFD simulations are lengthy, a feasible low-fidelity method is to be selected. The blade element theory enhanced with the finite state inflow model (**FISUW**) does not capture any vortex interactions, thus is not valid for anhedral optimizations. Yet, it is robust and represents the other design parameters fairly well during the optimization. It is preferred over the wake models, as the responses from these models are strongly amplified in this flight condition for some of the rotor designs found and erroneous results are retrieved, refer to Figure 3.12(a) and Figure 3.13.

During the sensitivity study as well as during the optimization, the required power computed by the wake models is very close to or even below the solution of the momentum theory, which features the theoretical minimum. This leads to strong doubts when practically applying these methods for an optimization. As a mid-fidelity model, inviscid CFD (**EU**) computations capture the important physical aspects well, while the run times are about half compared to viscous solutions on coarse meshes (**NS**). As the **EU** method has less overall dissipation than the **NS** method it captures the tip vortices better. For the high-fidelity method in hover, the RANS solution on the fine periodical meshes is assumed to be the best available solution. It is fairly costly during the optimization relative to the other methods, but also the most trustworthy solution. A summary of the run times and the equivalent number of low- and mid-fidelity computations, which could be performed instead, are listed on the left side of Table 3.7 for hover.

In forward flight the solutions of the low-fidelity methods are generally better than in hover. The solutions generated from the blade element theory coupled with a prescribed wake model (**P.WAKE**) are promising and the found optimum is even slightly better than the one found by the **EU** method. However, as this method only allows a limited number of discretization points, due to stability issues, specific parametrizations may not be resolvable with this method. Therefore, the inviscid single-blade Euler solutions are picked to be the mid-fidelity method, which have a very good agreement with the RANS solution for the baseline computation. Yet, this is based upon the assumption that friction can be neglected as well as the tip vortices in this flight condition. The question, which model is actually more suitable for supplementing the **FNS** method is still to be answered. The **P.WAKE** method is definitely cheaper than the **EU** method, thus it is preferable if this method turns out to be a better substitute for **FNS**, but from the type of physical model it is very different from **FNS**. While **EU** and **FNS** compute a field solution, the **P.WAKE** method computes vorticity potentials around the rotor. The multi-blade Chimera setup including the viscous solution of RANS equations (**FNS**) is considered the best-fidelity available, which then also completes the selection of fidelities in forward flight summarized in Table 3.7 on the right side.

Fidelity	**hover**	**forward flight**
low	**FISUW** 8 cpus	**P.WAKE** 1 cpum
mid	**EU** 5 cpuh ≈ 2,250 **FISUW**	**EU** 75 cpuh ≈ 4,500 **P.WAKE**
high	**FNS** 160 cpuh ≈ 32 **EU** ≈ 72,000 **FISUW**	**FNS** 2600 cpuh ≈ 35 **EU** ≈ 157,500 **P.WAKE**

Table 3.7: Choices of low-, mid- and high-fidelity for each flight condition and their CPU times (cpus=cpu seconds, cpum=cpu minutes, cpuh=cpu hours)

4 Detailed Analysis of Variable-Fidelity based Optimizations

Purpose of this chapter is on the one hand to prove that the variable-fidelity approach helps to speed up the optimization, while on the other hand the limitations of the approach are identified.

Two types of rotors are optimized with two individual parameterizations in two flight conditions each. The first rotor is the 7A rotor, which has been considered in the previous chapter. The set of four parameters remains unvaried, thus a comparison with the results from the previous chapter are easily made.

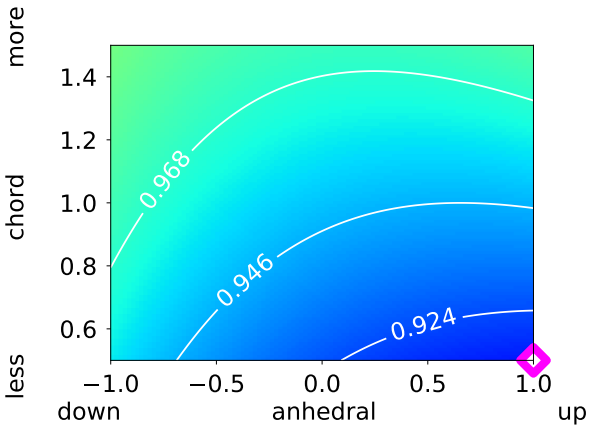
While the 7A rotor is a four-bladed model rotor from the mid-90s, the second rotor is the reference rotor of the Green Rotorcraft Research (GRC) project, part of the European JTI CleanSky project [149]. This rotor features five, instead of four blades. Its blade design represents a state-of-the-art configuration from current industrial research and thus a suitable test case for optimizing a full-scale rotor with variable-fidelity methods. For the GRC blade a ten dimensional parameterization is set up, Figure 4.20(a). In the first case, no constraints are considered, while in the second case, structural constraints are introduced to drive the resulting design more towards practical application.

4.1 7A Model Rotor Optimization with Four Parameters

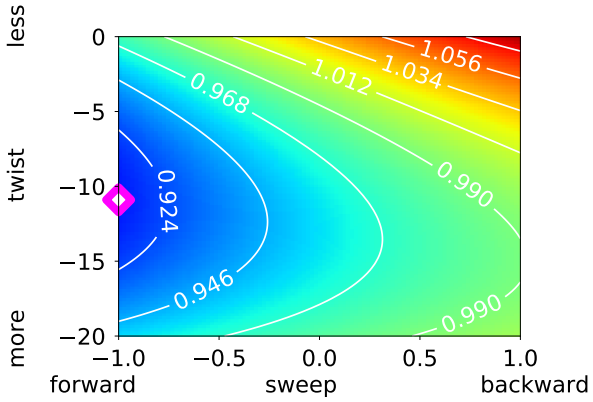
Goal of this optimization is to investigate the strength and weaknesses of the variable-fidelity approach for a small bounded parameter set without any constraints. Therefore, multiple test procedures are proposed. At first, the performance landscapes of the two goal functions, hover and forward flight performance, are examined in order to sensitize the reader to the problem at hand. With this knowledge the benchmark of the variable-fidelity approach with the single-fidelity approach is presented for single objective optimizations as well as multi-objective optimizations. The employed parameterization is identical to the one in Chapter 3.3, see Figure 3.2.

4.1.1 Introduction to the Performance Landscapes

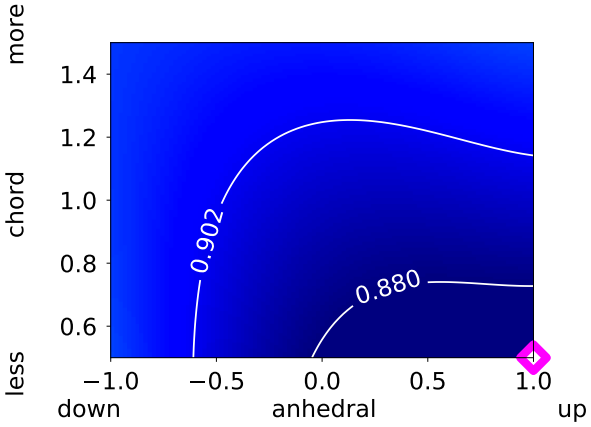
Before starting with the benchmarking of the single- and variable-fidelity optimizations, the individual goal functions are analyzed on a deeper functional level **without** a detailed interpretation of the underlying physics. It is an extension of the sensitivity



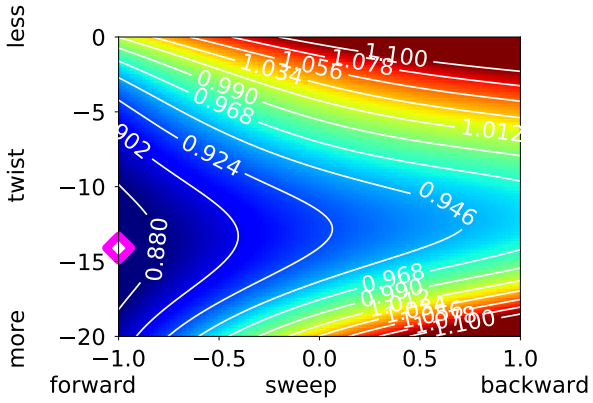
(a) **FNS** (HiFi): anhedral-chord slice



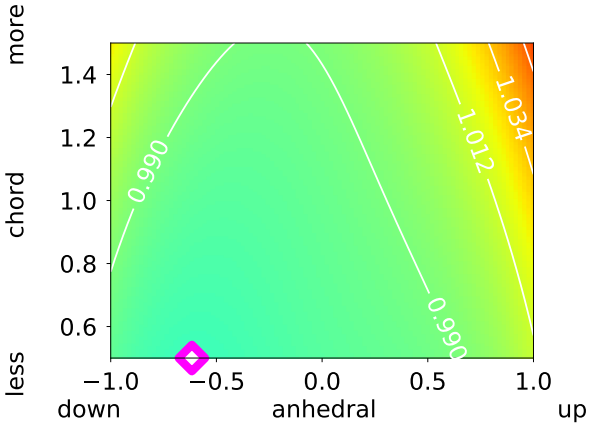
(b) **FNS** (HiFi): sweep-twist slice



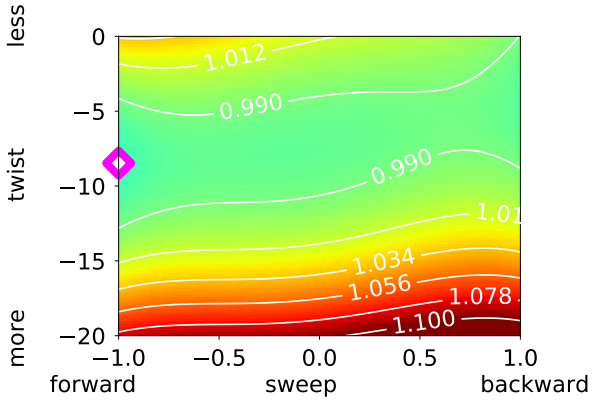
(c) **EU** (MiFi): anhedral-chord slice



(d) **EU** (MiFi): sweep-twist slice



(e) **FISUW** (LoFi): anhedral-chord slice



(f) **FISUW** (LoFi): sweep-twist slice

Figure 4.1: Sketch of the **global hover minimum** of power fraction for each-fidelity of the 7A rotor. Same coloring for same values. The magenta diamond denotes the minimum (◊).

study in Section 3.3, but instead of investigating the sensitivity of the baseline rotor, the design space around the optima predicted by the individual methods is explored.

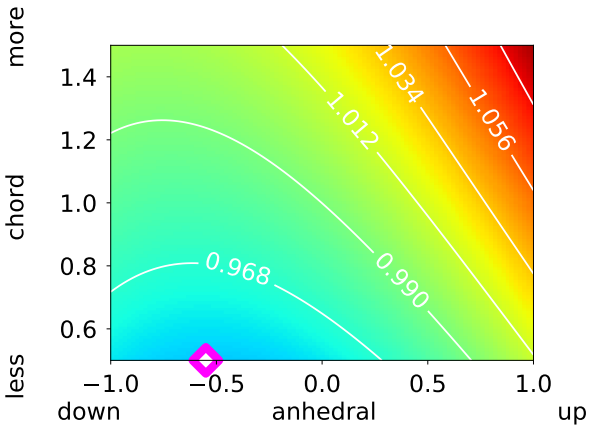
Unlike the synthetic test function in Section 2.6, no direct solution exists for the required power of the helicopter rotor. Therefore the surrogate models from the later on performed optimizations are exploited here to draw the performance landscapes. The idea is to explain to the reader the similarities and differences of the goal functions given from the individual aerodynamic simulations. The design space is presented in multi-dimensional slices. The principle behind this slicing is explained in the appendix in Section A.1.3. The parameter space represents a four-dimensional hypercube defined by the parameters anhedral, chord, sweep and twist. The visualization is done in two-dimensional cuts with the color contour being the power fraction.

In Figure 4.1 the landscapes for the individual global hover optima of each fidelity are sketched. On the left, the anhedral-chord plane is given, while on the right side the sweep-twist plane is plotted. The magenta diamond marks the individual global optimum (◊), also being the location at which the slices are made. Thus, this diamond couples the left with the right picture. Blue denotes areas of improvement, red areas are worse than the baseline rotor and green areas are the transition between these extremes. On top, the high-fidelity (**FNS**) is depicted, in the middle the mid-fidelity (**EU**) and at the bottom the low-fidelity (**FISUW**). The scaling of the colors and the isolines are kept the same, thus not only the location of the minima is visible, but also their magnitude. For the nomenclature of the methods, please refer to Table 3.4.

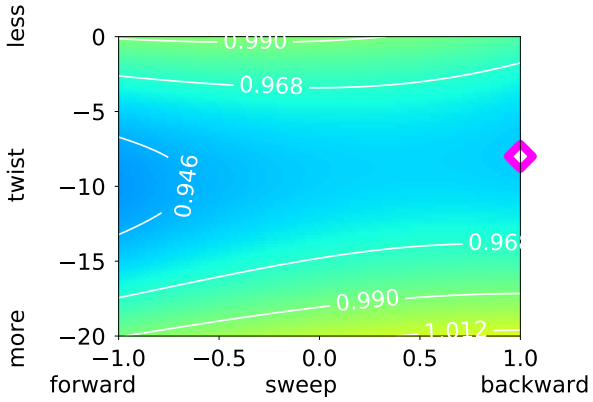
From Figures 4.1(a) and 4.1(b) it is seen that the shape of the high-fidelity landscapes features a shallow area around the optimum in both planes. The mid-fidelity on the other hand has a rapid increase of the twist gradient if one moves away from the mid-fidelity optimum, Figure 4.1(d), but the anhedral-chord plane is even more shallow than it is for the high-fidelity Figure 4.1(c). The position of the mid-fidelity optimum mostly matches with the one from the high-fidelity with the exception that the (negative) twist is slightly larger. A certain resemblance of the high-fidelity landscape is observed.

Moving onto the low-fidelity landscape, Figures 4.1(e) and 4.1(f), it is recognized that this function has less similarities than the mid-fidelity with the global optimum of the high-fidelity, but shares some common attributes. The chord and twist parameter have a similar position of the optimum as well as trend. The required power linearly decreases with the chord length, while the parabolic behavior in dependency of twist is also given. The sweep parameter introduces a second minimum in the sweep-twist plot, Figure 4.1(f). This has also been discovered in the sensitivity study, Figure 3.12(a), for the baseline rotor for all examined fidelities.

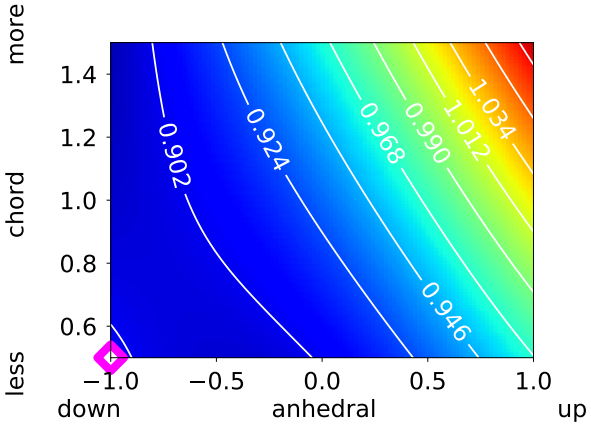
This local hover minimum is assumed to exist for the mid- and high-fidelity derived from the sensitivity study in Section 3.3, Figure 3.11(a). There the sweep parameter shows that either forward or backward sweep is preferable. With the aid of the surrogate model, this second local minimum is discovered for each fidelity. The according landscapes are plotted in Figure 4.2. For all three fidelities, the second minimum is best found in the sweep-twist plots (Figure 4.2(b), Figure 4.2(d), Figure 4.2(f)). On the left side is the forward sweep, where for all three fidelities the direction towards the global minimum is eminent. On the right side, the backward sweep is given and the local minimum is noticed. The minimum is most distinct for the **EU** function, while for the **FNS** as well as **FISUW** functions the local minimum is less pronounced, but existent. Comparing this with the plots of the global optimum in Figure 4.1, it is concluded



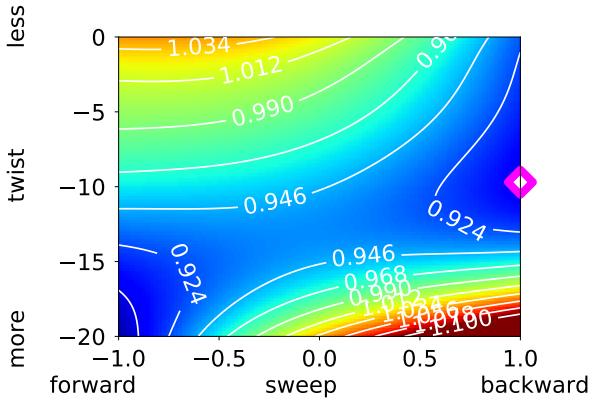
(a) **FNS** (HiFi): anhedral-chord slice



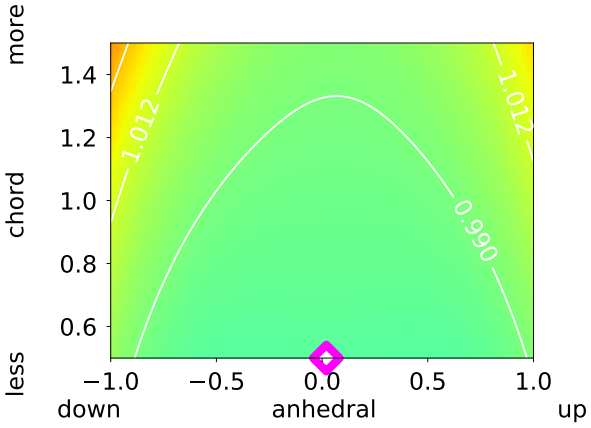
(b) **FNS** (HiFi): sweep-twist slice



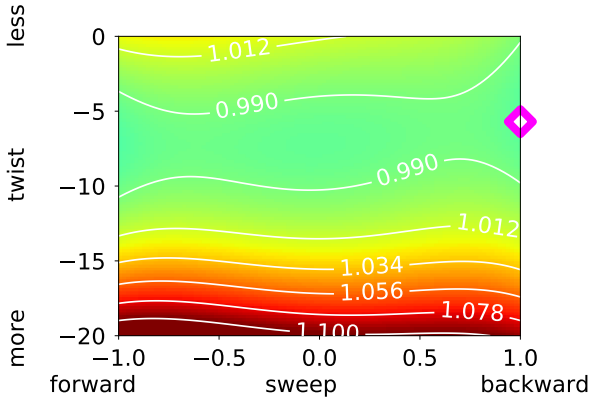
(c) **EU** (MiFi): anhedral-chord slice



(d) **EU** (MiFi): sweep-twist slice



(e) **FISUW** (LoFi): anhedral-chord slice



(f) **FISUW** (LoFi): sweep-twist slice)

Figure 4.2: Sketch of the **local hover minimum** of power fraction for each-fidelity of the 7A rotor. Same coloring for same values. The magenta diamond denotes the minimum (◊).

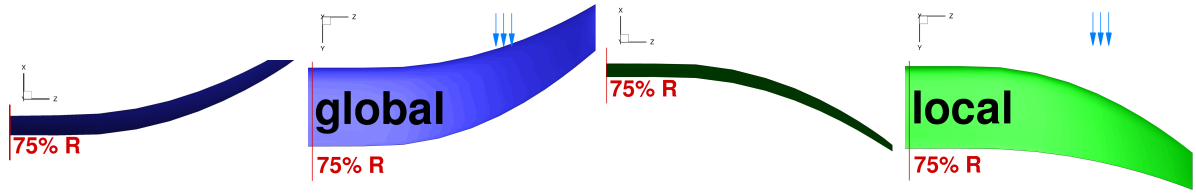


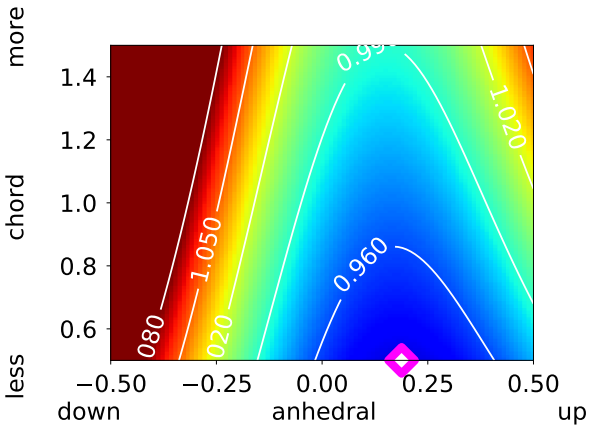
Figure 4.3: Blade tips of the global (left) minimum and the local minimum (right) from the mid-fidelity hover surrogate model for 7A rotor.

that the hover goal function is strongly non-linear. The low-fidelity only exhibits little change when going from the local to the global minimum, but the mid- and high-fidelity change the location of anhedral and sweep to the opposite side in the design space. To comprehend the magnitude of this functional relation, the corresponding designs of the global and local hover minimum of the mid-fidelity function are pictured in Figure 4.3. This local minimum re-appears in the results of the multi-objective optimization in Section 4.1.4

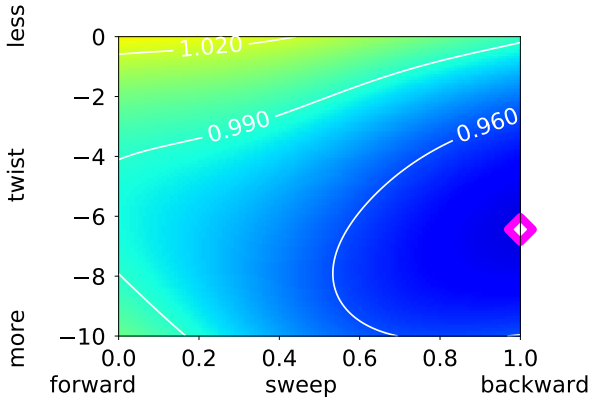
Deducting from the discussion of the hover goal function landscape, the following conclusions are drawn; the (visual) correlation of the low-fidelity to high-fidelity is smaller than the mid- to the high-fidelity. Nevertheless, the low-fidelity function shares some attributes with the mid- and high-fidelity function. The linear behavior of the chord parameter is reflected by the low-fidelity, as well as the parabolic trend of the anhedral, sweep and twist parameter. The strong difference to the mid- and high-fidelity arises from the anhedral and sweep parameter, which are less coupled for the low-fidelity and is attributed to the lack of being able to model three dimensional flow effects. The mid- and high-fidelity have more in common, though the mid-fidelity predicts greater improvements, has a stronger growth of the twist gradient when going away from the optimum and has a more pronounced local minimum than the high-fidelity.

The forward flight goal function is pictured in Figure 4.4. Since forward sweep is not considered in this flight condition due to the strong aero-elastic coupling, no second minimum is found.

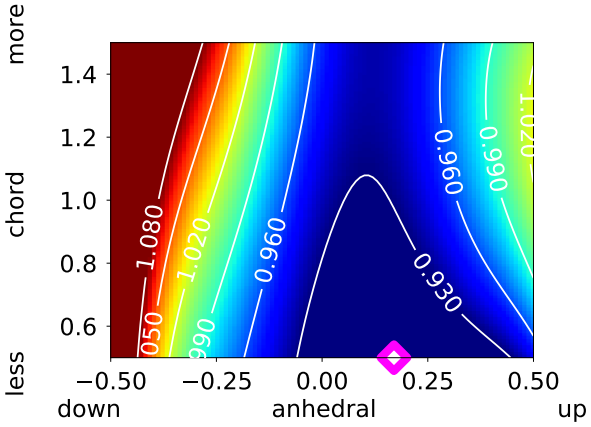
A good agreement among the various methods is observed in forward flight. Not only the shape matches better along all three fidelities compared to the hover results, but also the attainable level of improvement, even though the high-fidelity features less improvement than the mid- and low-fidelity. In contrast to the hover function, the anhedral and sweep gradient is stronger when moving away from the optimum, while the chord parameter has a more benign behavior. This also matches with the findings from the investigation of the sensitivity study of the previous chapter. Overall, it must be said that for forward flight the low- as well as mid-fidelity are equally good in resembling the high-fidelity leading to the assumption that triple-fidelity optimization setups or low-high-fidelity combinations might work just as well as mid-high-fidelity combinations in forward flight.



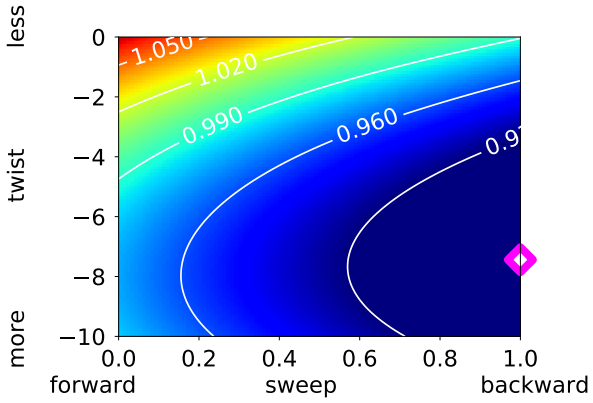
(a) **FNS** (HiFi): anhedral-chord slice



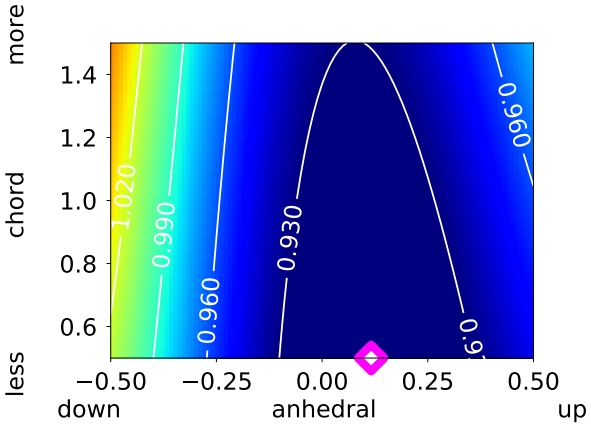
(b) **FNS** (HiFi): sweep-twist slice



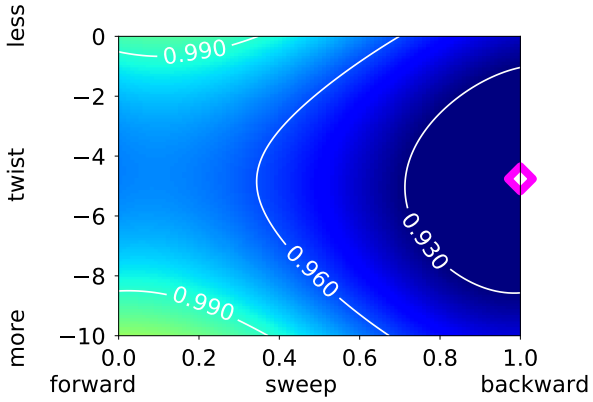
(c) **EU** (MiFi): anhedral-chord slice



(d) **EU** (MiFi): sweep-twist slice



(e) **P.WAKE** (LoFi): anhedral-chord slice



(f) **P.WAKE** (LoFi): sweep-twist slice

Figure 4.4: Sketch of the **global forward flight optimum** of power fraction for each-fidelity of the 7A rotor. Same coloring for same values. The magenta diamond denotes the minimum (◊).

The major points of the performance landscape examination are summed up:

- The hover goal function contains two minima, which are mostly governed by the sweep parameter.
- In hover, the low-fidelity model **FISUW** correlates less with the high-fidelity function **FNS** than the mid-fidelity **EU** does.
- The forward flight design space has only one minimum as the aero-elastic coupling forbids forward sweep. The parameter coupling is also weaker than with hover.
- In forward flight, the different fidelities align well. **P.WAKE** mostly differs from **FNS** in the trend of the chord and sweep parameters, while **EU** show differences in the chord and twist parameter.

4.1.2 Test Matrix

The combinations of employed methods for the 7A variable-fidelity optimization studies are depicted in Figure 4.5. The mid-fidelity tests are performed with **FISUW** in hover and **P.WAKE** in forward flight as the low-fidelity. Both flight conditions are coupled with **EU** simulations as the highest fidelity for this mid-fidelity test. The high-fidelity optimization employ the **FNS** method for hover and forward flight. Besides single-objective optimizations, multi-objective optimizations are also undertaken. When optimizing at high-fidelity, various constellations are tried such as mid-high, low-high and triple-fidelity setups.

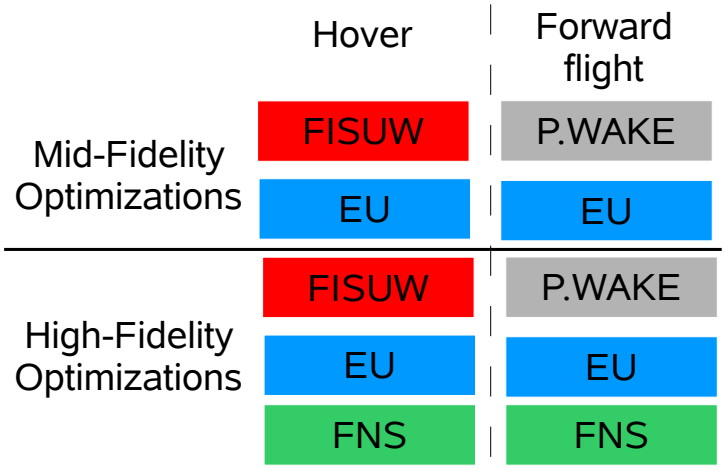


Figure 4.5: Test matrix of 7A rotor optimization

The order of presentation of these tests is as follows:

1. Single-Objective (hover and forward flight individually)
   (a) Mid-Fidelity Optimizations
   (b) High-Fidelity Optimizations
2. Multi-Objective (hover and forward flight together)
   (a) Mid-Fidelity Optimizations
   (b) High-Fidelity Optimizations

The single-objective is discussed first to gain knowledge of the behavior of the individual goal functions, before both goal functions are simultaneously optimized.

4.1.3 Single-Objective Optimization

The single-objective optimizations are presented first as the results are easier to understand and the benefits for each goal function utilizing the variable-fidelity approach are quantified. The mid-fidelity optimizations either employ a single-fidelity setup or a variable-fidelity setup, but their highest fidelity level is the **EU** method for both cases. The high-fidelity optimization then end up at **FNS** level and take up to three fidelity levels in total to perform the optimization.

Mid-Fidelity Optimizations

Due to the short turn-around times, the mid-fidelity allows to investigate more settings for the surrogate based optimizations, which are then applied to the high-fidelity optimization. The downside is that triple-fidelity setups cannot be tried. Strategy 1

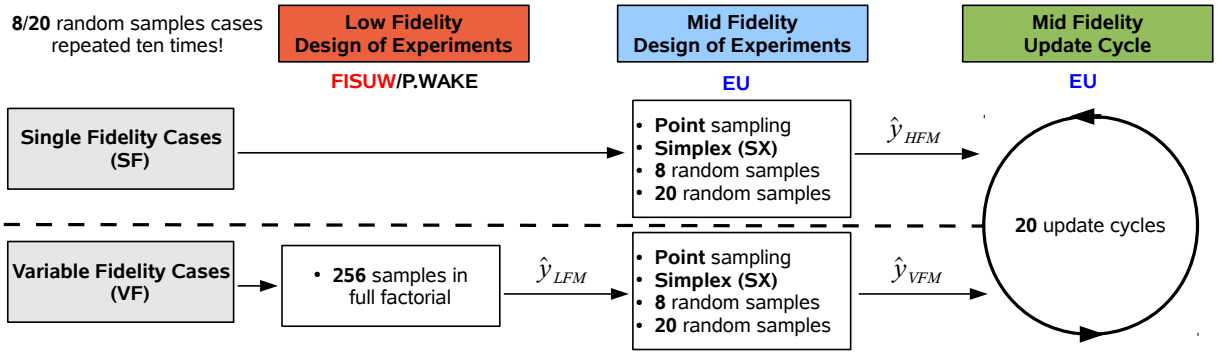


Figure 4.6: Test setup for the 7A mid-fidelity single-objective tests in hover and forward flight.

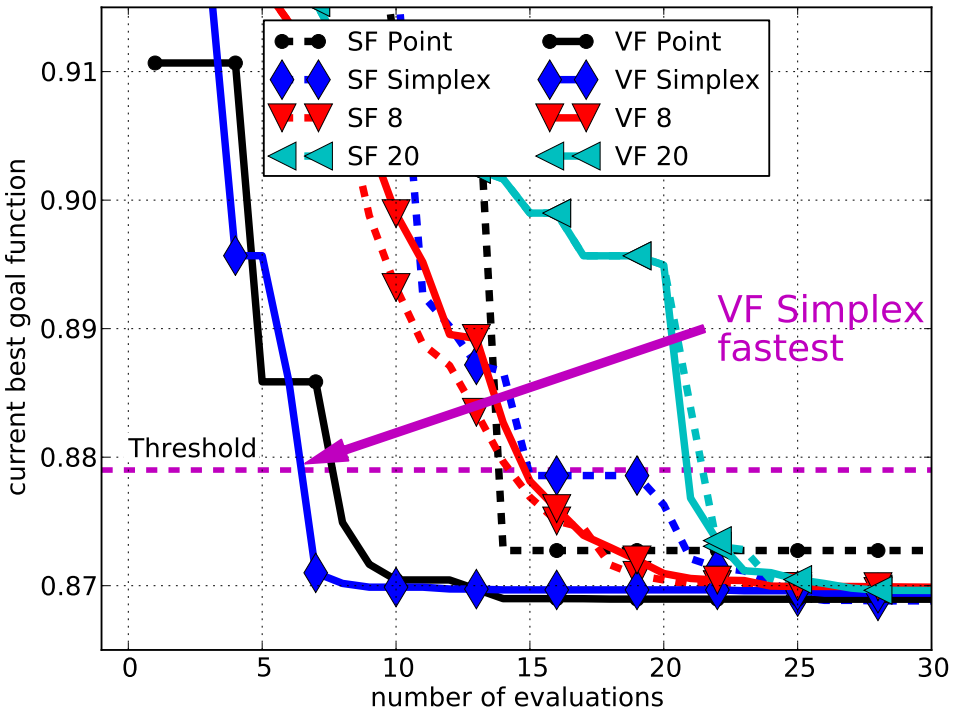
In Figure 4.6, the two examined fidelity modes, single- and variable-fidelity, along with their parameter combination are sketched. The naming convention is similar to the one in Section 2.6. With the help of Figure 4.6, the investigated single-fidelity cases are then **SF Point**, **SF Simplex**, **SF 8**, **SF 20**, the according variable-fidelity cases are referenced by **VF**. The **Point** and **Simplex** design of experiments are either a single-point or a simplex placed at this single point. This point corresponds to the 7A rotor

for the single-fidelity optimizations, while for the variable-fidelity this is the optimum found in the low-fidelity model. The **8** and **20** design of experiments are randomly generated hypercubes, as explained in Section 2.2. These cases are repeated ten times with different random hypercubes to later on perform statistics on the results.

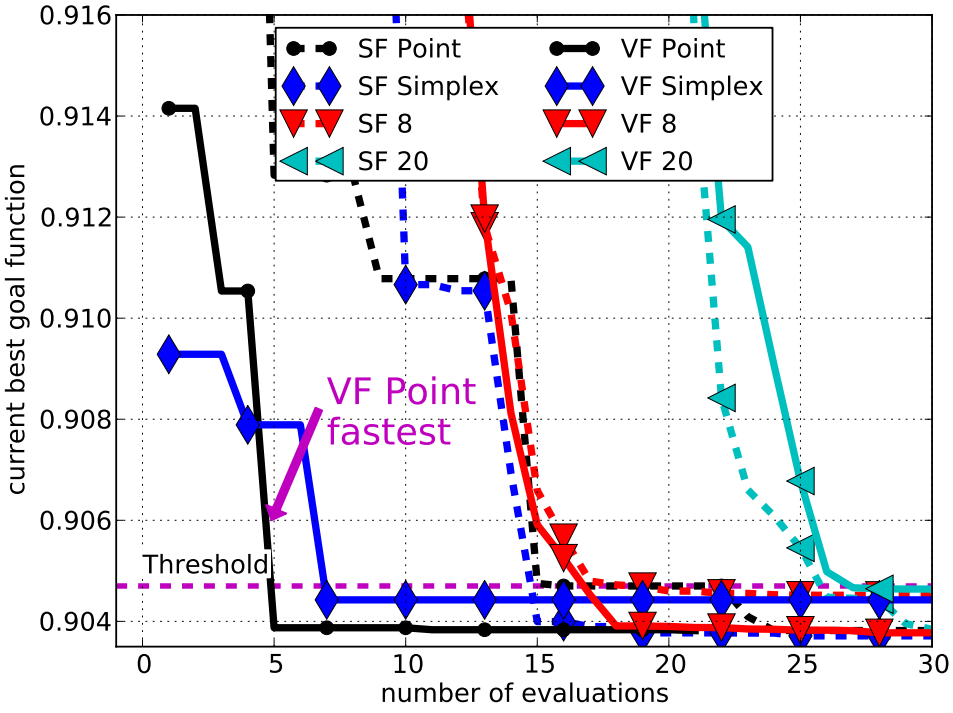
The numerical settings for the surrogate models are a search range for $\theta = [10^{-4}, 10^{-1}]$, a set λ of 10^{-10} for the low-fidelity and 10^{-15} for the mid-fidelity. The deterministic optimization **Strategy 1** from Section 2.4.1 is employed for the search of the highest expected improvement as well as the optimization of the hyper parameters of the Kriging models.

A graphical representation of the convergence of the optimizations is given in Figure 4.7(a) for hover and Figure 4.7(b) for forward flight. The dashed magenta horizontal line marks the threshold (--), which is used to determine the cost of each optimization. The threshold is determined aposteriori, mostly based upon the best configuration found by all given runs and adding a specific allowable margin to it. For this scenario, no stopping criterion is enforced and all cases are allowed to converge. However, the design confidence stopping criterion is computed, which would stop the optimization if with 90% certainty the best surrogated point improves less than 0.01, see Section 2.3.4 for more details on design confidence. The dashed lines (--) represent single-fidelity results, while the variable-fidelity has the continuous lines (—). The lines reflect the currently best sampled result, not the result of the current sample. The expected improvement sampling searches the whole design space and does not necessarily improve the goal function during each iteration. Similar setups feature similar markers and lines, so that they are easily compared. In simplified terms, in order to prove that the variable-fidelity approach is superior to the single-fidelity approach, the continuous lines (—) have to drop below the threshold (--) before the dashed lines (--) do. For this scenario, the cost for the low-fidelity is negligible and thus only the number of mid-fidelity evaluations is counted as the overall cost of the optimization.

Beginning with the hover results, the threshold for the hover goal function value is set to 0.879, 0.015 better than the best sample found by a run of the **VF 20** case. A first positive statement is that all optimizations are faster in comparison with the genetic optimization undertaken in Section 3.3. The genetic hover optimization required a total of 290 simulations, the most costly surrogate based optimization required 21 simulations, while also maintaining a higher accuracy meaning that the goal function values are smaller. All goal function values of the final configurations are below 0.870 with design vectors similar to the one of the genetic optimization performed in Section 3.3. This leads to the conclusion that all optimizations found the global optimum within the allowed margin of 0.015 of the goal function value employed for the design confidence criterion. From the plot in Figure 4.7(a), the goal that the continuous lines are below the dashed ones is mostly fulfilled and the variable-fidelity methodology is faster in most cases. However, the **SF 8** and **VF 8** case have a very similar convergence. This has two reasons. Firstly, this type of hypercube employed for these cases does not include any information from the low-fidelity itself in terms of the selection of points like the low-fidelity optimum, while secondly the number of mid-fidelity samples is relatively large. Thus, the low-fidelity trend is surpassed by the error correction term from the radial basis functions. The hypothesis is strengthened when looking into Figure 4.8(a), where the **SF 20** and **VF 20** cases are additionally presented. This chart contains the costs to reach the specified threshold value (filled area) along with the first time the design confidence reaches more than 90.0 % for a difference of 0.01



(a) Hover (threshold is 0.879)



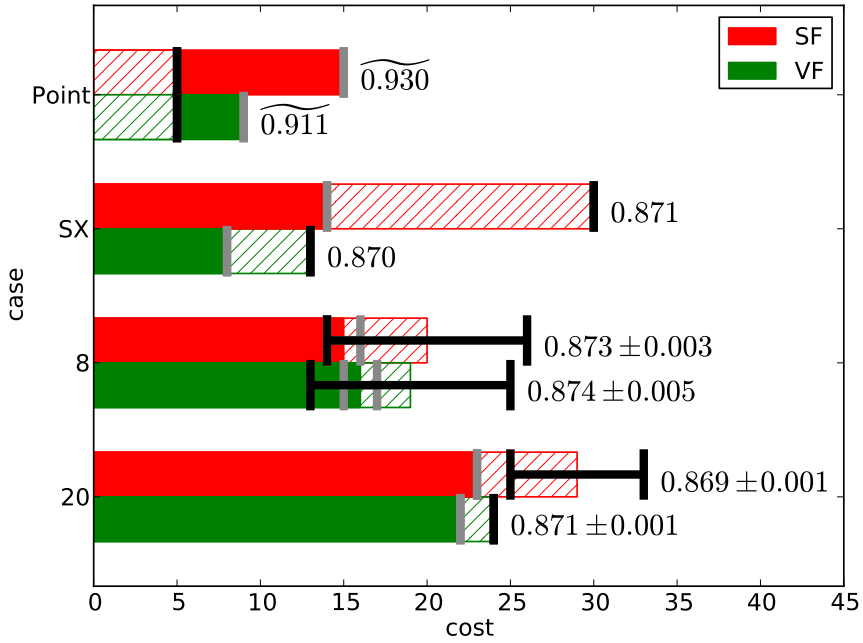
(b) Forward flight (threshold is 0.905)

Figure 4.7: 7A single-objective optimization at **mid-fidelity**: Convergence of single- (SF) and variable (VF)-fidelity for single-objective. Goal function values averaged over ten runs.

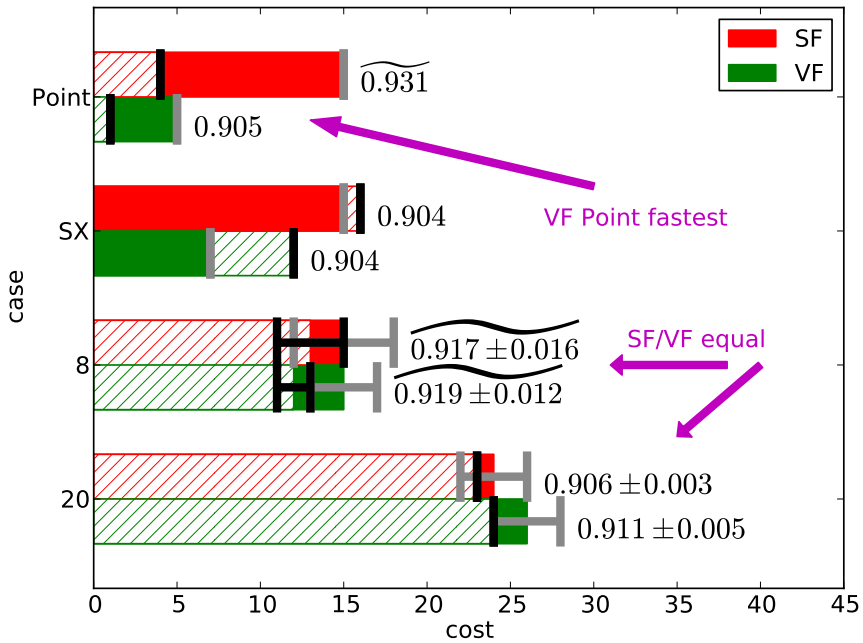
(hatched area) and the best goal function value sampled when reaching the design confidence. The design confidence is a desirable stopping criterion, which tells the blade designer, when the currently found minimum is actually the global optimum within a given probability and threshold tolerance. Thus both costs are mentioned here. On academic terms, reaching the threshold is the metric for comparing the efficiency of single- to variable-fidelity optimizations, while when an engineer would invest the additinal cost for the design confidence to be sure the optimization is finished. If a tilde is above the goal function value, the current best sample is further away from the optimum than theoretically allowed by the design confidence criterion. For these cases, the hatched and filled areas are flipped to indicate that the optimal configuration is found after the optimization process reached certainty it found the optimum. The gray error bars in the chart visualize the standard deviation of iterations required for reaching the threshold, while the black error bars visualize the standard deviation of iterations required to reach the design confidence. The standard deviation of the goal function value is added to the mean goal function value with the $\pm$ sign. The **SF 20** and **VF 20** cases reach the threshold during their second and third update iteration leading to similar costs for both methods. The low-fidelity neither helps nor slows down the optimization at this point, because the mid-fidelity design space is abundantly sampled. Only the cost for reaching the design confidence is reduced through the variable-fidelity approach in this scenario.

The next observation is that the point samplings find their design confidence too early. If the stopping criterion of the design confidence was applied, the optimization would stop prematurely thinking it would be done. The reason for this is that with the little amount of points given, the update cycle starts sampling the corners of the hyperspace, because these are furthest away from the initial point and the statistical error from the Kriging model is the largest. The variable-fidelity places the first three samples in corners of the hypercube, while the single-fidelity goes for nine of the 16 corners of the design space before the expected improvement also acts upon the goal function value. The problem here is that all corner points feature goal function values that are worse than the initial design. For the single-fidelity, with four points or less in the surrogate model, the trend function is a constant with four radial basis functions super positioned over it. Three of those basis functions point upward (higher goal function values), while the one basis function belonging to the initial sample points downwards (lower goal function value). When the design confidence is computed, the best configuration is searched in the surrogate model, which then only points to the initial sample resulting in a high design confidence as the error for this point is zero per definition (for $\lambda = 0$). Thus, the design confidence is tricked if not sufficiently enough samples are in the design of experiments, as the initial function approximation becomes very simplified and does not resolve important features of the performance landscape. This can also occur with the variable-fidelity method. In particular if the low-fidelity is scaled in such an odd way by the tuning process that it has diminishing influence on the prediction. This also demonstrates that the statistical value of the design confidence has to be used with caution. Nevertheless, if the update cycle is continued, as done here, the surrogate based optimization process still converges and for larger design of experiments, the design confidence is plausible.

Moving onto comparing the single to variable-fidelity optimizations in hover; the variable-fidelity approach is superior, but for a sufficiently large initial design of experiments the advantage diminishes. This has also been recognized in Section 2.6 for the synthetic test function. The most efficient case for reaching the threshold is the **Simplex** design



(a) Hover (threshold is 0.879)



(b) Forward flight (threshold is 0.905)

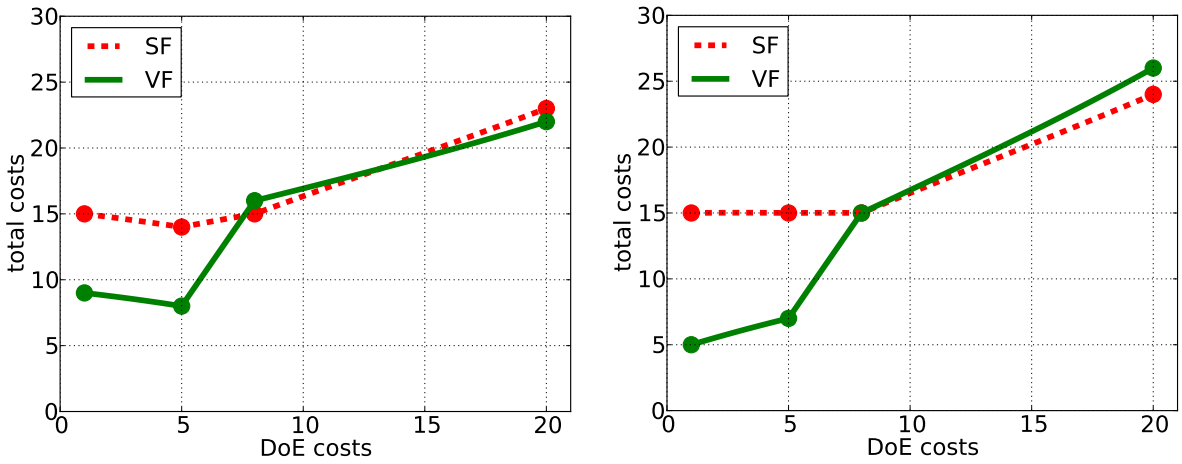
Figure 4.8: 7A single-objective optimization at **mid-fidelity**: Cost to reach either threshold (filled area) or design confidence (DC) criterion (hatched area) including standard deviation bars. The numbers are the goal function values upon reaching DC with their respective standard deviations. SF = single fidelity, VF = variable fidelity. Results with a tilde (˜) highlight that the goal function value is above the 0.01 margin for the DC criterion.

of experiment for both types of optimization followed by the simple **Point** design of experiment. For reaching a valid design confidence, the **SF 8** case is the best for the single-fidelity and the **simplex** sampling strategy for the variable-fidelity. Comparing the most efficient cases from both fidelity modes, the variable-fidelity requires 46.2% less mid-fidelity samples to arrive at the optimal configuration measured in number of samples required to reach the threshold. The variable-fidelity spends 42.9% less samples than the single-fidelity to be confident that this configuration is not superseded by a new sample within the 0.01 margin

Last but not least, the effect of the randomization is small, yet given. It is defined through the standard deviation, which represents the expected interval around the mean value. Larger standard deviation means that the chances are greater that the optimization finishes earlier or later, while small standard deviations stand for a greater reliability of the optimization process. Especially for the smaller design of experiment with only eight samples, the standard deviation of the finishing iterations is larger than for the design of experiments with 20 samples. While no difference in the standard deviation is noted for finding the optimum between single- and variable-fidelity, the standard deviation decreases for the design confidence by using the variable-fidelity approach.

Continuing with the forward flight optimization, Figure 4.7(b), it becomes apparent that the surrogate based optimization process is highly beneficial for this optimization problem. While the genetic optimization process in Section 3.3 required 290 simulations, the worst surrogate based optimization, a **VF 20** case, required 26 samples. The margin of the threshold for comparison is chosen to be 0.905, which is only 0.0015 worse than the best found configuration, due to this benign optimization problem arising from the bounded parameter space for the forward flight optimization. During the optimization the margin for the design confidence is set to 0.01, which would allow the optimization to stop after reaching a value of 0.915 or below. Looking at the tendencies of the individual cases, similar effects noticed in the hover scenario are also found for forward flight. The point sampling achieves design confidence prematurely, which is concluded from Figure 4.8(b). For larger design of experiments the underlying trend of the Kriging function becomes less important as observed during the hover optimization. The gain of the variable-fidelity process pays out for smaller design of experiments, when reaching the best configuration is wanted. The **VF Point** test requires one third of the resources in comparison with the **SF Point/Simplex/8** cases, which all required 15 samples in total on average to reach the optimum. However, reaching the design confidence is not as cost saving. If the **VF Point** case is left out of this consideration as it reaches the design confidence before the actual optimum, then the benefit is barely measurable with 12 samples for the **VF Simplex** case in relation to the **SF 8** case with 13 samples in total. Apparently a certain number of samples is required to drive the overall error of the surrogate models below a certain limit in order to obtain a sufficient design confidence. Going back to the hover simulation, the minimum number of samples to reach the design confidence is 12, which is roughly the same magnitude. Nevertheless, if the **VF Point** case is considered as valid, the cost savings would be 66.7% for sampling the optimal configuration and 69.2% for the design confidence.

As for the randomization effect, it is greater for forward flight than it is for hover. No distinct statement can be made whether the variable-fidelity optimizations reduces this or not.



(a) Hover (b) Forward flight

Figure 4.9: 7A optimization at **mid-fidelity**: Total cost to reach threshold over cost for design of experiments for single (SF) and variable (VF)-fidelity.

From these first investigations, four generalizations are made:

1. Variable-fidelity allows a faster optimization when the size of the design of experiments is well chosen. This is clarified in Figure 4.9(a) and Figure 4.9(b), where the total cost of the DoE costs is plotted. In these cases, for the smaller DoEs the variable-fidelity requires significantly less resources.

2. Large DoEs make the Kriging process independent of the trend function, which is also supported by Figure 4.9(a) and Figure 4.9(b). The total cost for larger DoEs is almost equal between single- and variable-fidelity.

3. The chance of the design confidence being erroneous grows for smaller number of samples in the DoEs.

4. Large DoEs reduce the randomization effect.

High-Fidelity Optimizations

The high-fidelity optimizations are performed with state-of-the-art CFD computations. As the turn-around times are significantly larger than for the mid-fidelity optimizations, only a compact set of combinations is investigated in contrast. On the one hand, triple-fidelity and low-high fidelity setups are investigated, while on the other hand no averaging over multiple runs is performed.

The designs of experiments are modified in reference to the previous mid-fidelity optimizations, and different sets of double and triple-fidelity cases are examined. The flowchart is presented in Figure 4.10. For the single-fidelity, three different design of experiments are tried. For the variable-fidelity, four cases (**VF Low, VF Mid 1, VF Mid 2, VF 3F**) sample the optimum of their underlying trend as their design of experiments. The design of experiments of the fifth variable-fidelity case contains eight random samples. The **VF Point Low**, the **VF Point Mid 1** and **VF 8 Mid 1** cases are enhanced with a low- or mid-fidelity surrogate model generated from a full factorial

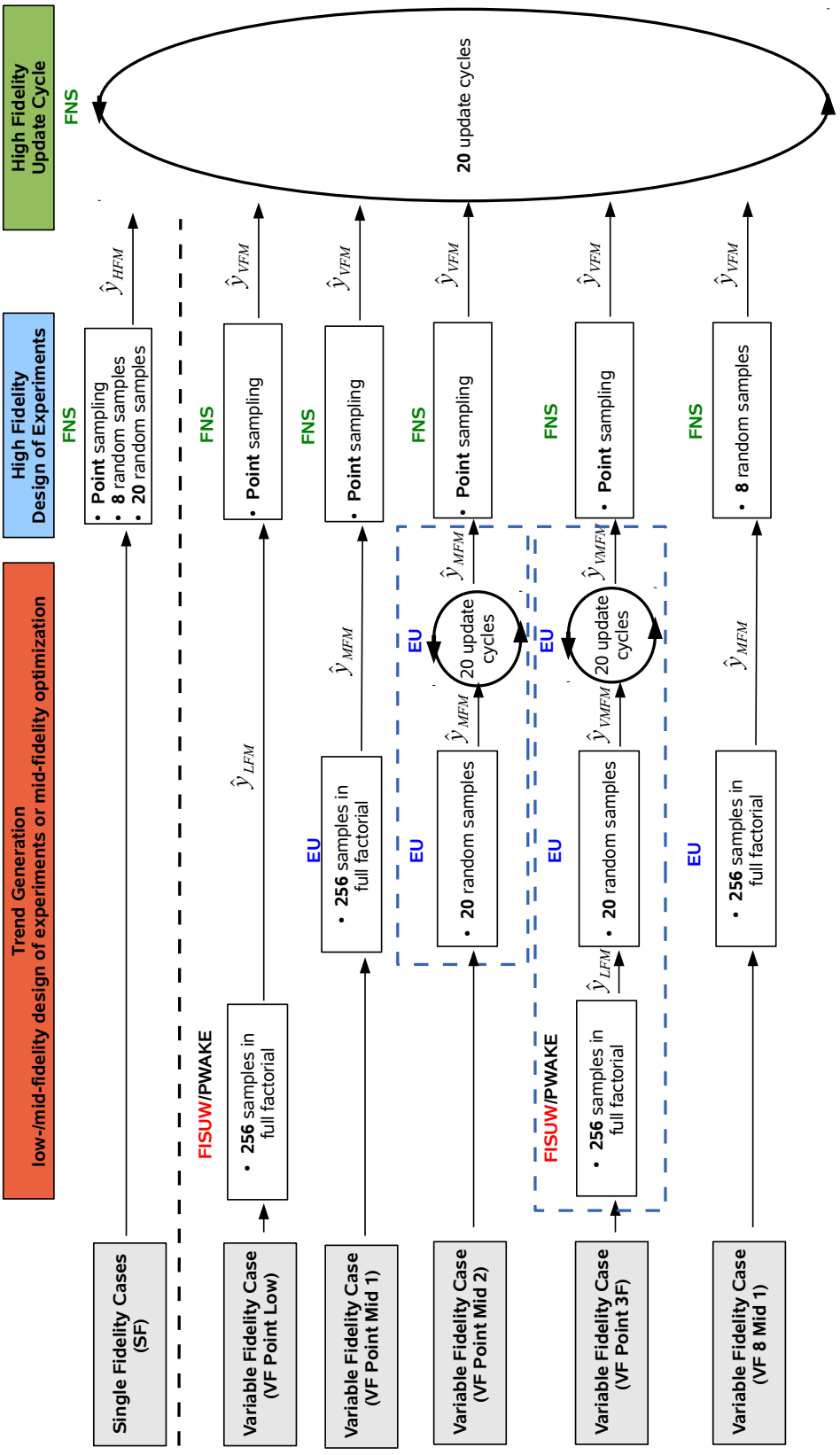


Figure 4.10: Test setup for the 7A high-fidelity single-objective tests in hover and forward flight. Dashed blue boxes represent the re-use of the mid-fidelity surrogate models, see Figure 4.7.

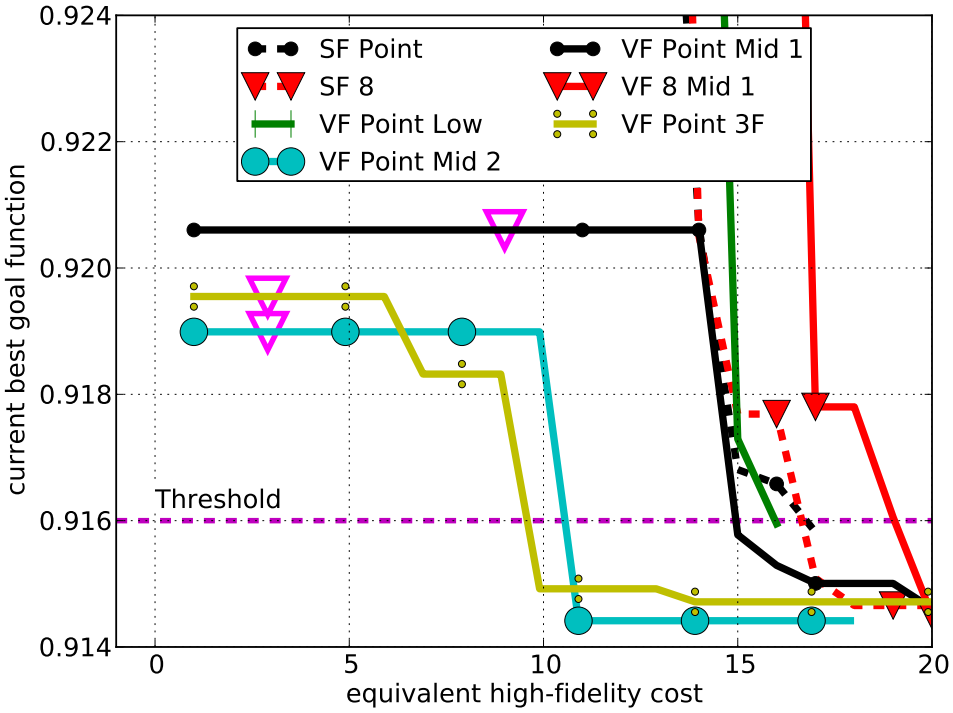
sampling. The **VF Point Mid 2** case recycles the surrogate model from the mid-fidelity investigation obtained by the best **SF 20** case. The **VF Point 3F** is the triple-fidelity setup, which recycles the best **VF 20** mid-fidelity case. All cases then run 20 update cycles. The dashed blue boxes of Figure 4.10 represent the data from the mid-fidelity optimizations from the previous section, refer to Figure 4.7.

The outcome of the optimizations of each flight condition is plotted in Figure 4.11(a) and Figure 4.11(b) for hover and forward flight, respectively. For these plots, the cost of the low- or mid-fidelity is included and counted as equivalent single-fidelity iterations at high-fidelity level. The magenta triangle marks the cost for the whole trend generation (∇). The here mentioned cost is determined with the help of Table 3.7 and represents the equivalent computational time of one high-fidelity simulation.

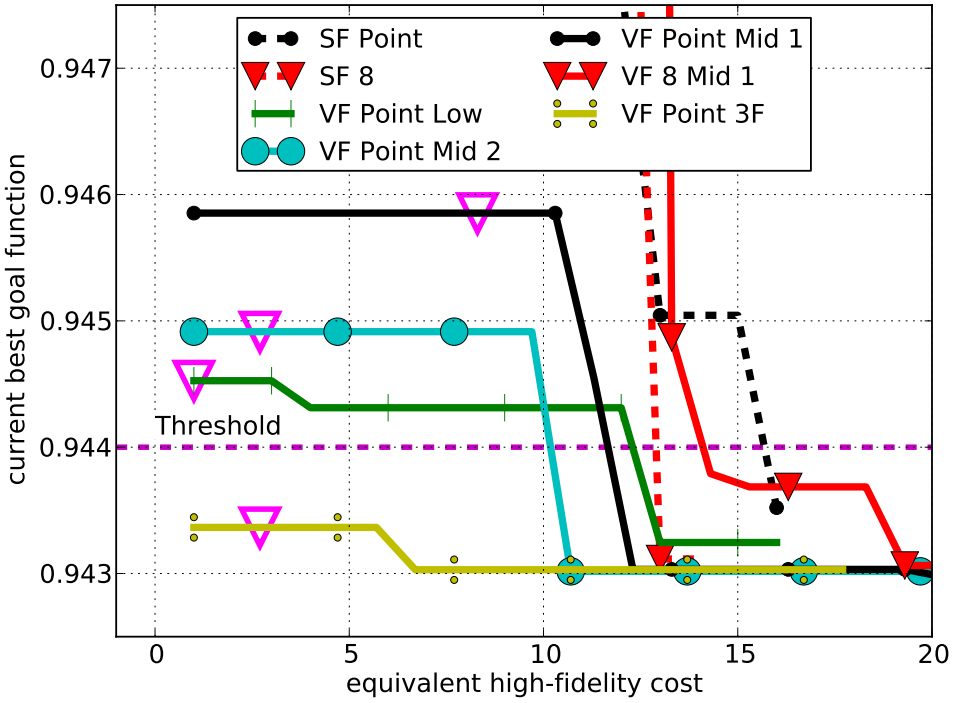
In hover, four out of five variable-fidelity setups prove to be faster than the best single-fidelity run, **SF Point**. The **VF 8 Mid 1** case finds the optimum and design confidence without problems, but the surrogate model is saturated and the design of experiments is chosen too large to be cost efficient overall. The **VF Point Low** case is faster by one high-fidelity sample in contrast to the **SF Point** case, a minor advance over the single-fidelity. It costs actually more to go from the low- to the high-fidelity than from the mid- to the high-fidelity, despite the fact that the mid-fidelity costs are not negligible anymore. This problem occurs from the fact that the low-fidelity has a too large offset from the high-fidelity, which is sketched in Section 4.1.1 of the performance landscapes. Correcting the misleading low-fidelity model is more costly than starting from scratch, similar to the example of the synthetic test function with the second low-fidelity from Section 2.6. The quickest process is the **VF Point 3F** case. The speed is attributed to the refinement of the surrogate model in the region of interest rather than the triple-fidelity setup, as the **VF Point Mid 2** case requires almost the same amount of resources to reach the threshold. This means that if the underlying trend is able to accelerate the optimization, it itself should be sufficiently accurate in the regions of interest as otherwise the coarse resolution in the low- or mid-fidelity need to be corrected by the high-fidelity. The cost more than doubles in this case when going from the locally refined mid-fidelity of **VF Point Mid 2** to the globally accurate **VF Point Mid 1** model; The latter turns out to be coarser in the region of interest despite the many points evaluated.

Benchmarking the efficiency of the single to the variable-fidelity in hover, Figure 4.12(a) visualizes the raw numbers. It is recognized that the two most cost efficient setups are the **VF Point Mid 2** and **VF Point 3F** cases, which require an equivalent of 10.9 and 9.9 high-fidelity samples to reach the optimum. In contrast to this, the **SF Point** and **SF 8** cases need 17 samples. In terms of efficiency, the variable-fidelity setup required about 58.2% of the computational effort (or reduced it by 41.8%). A similar effect from the mid-fidelity forward flight optimization is observed for evaluating the cost to reach the design confidence. Apparently a sufficient number of samples in the high-fidelity surrogate is necessary to bring down the error predictor of the Kriging model and thus to grant the required certainty. The **SF Point** and **VF Point Low** prematurely achieve their design confidence, the hatched area ends before the filled area, thus they are not considered for comparison. The **SF 8** and **VF Point Mid 2** require 17 and 5.9 high-fidelity sample equivalents, which corresponds to a reduction of 65.3% for the variable-fidelity optimization.

In Figure 4.12(b), the costs of the individual forward flight cases are given. The best run is the **VF Point 3F** followed by the **VF Point Mid 2**. The cost for finding the op-

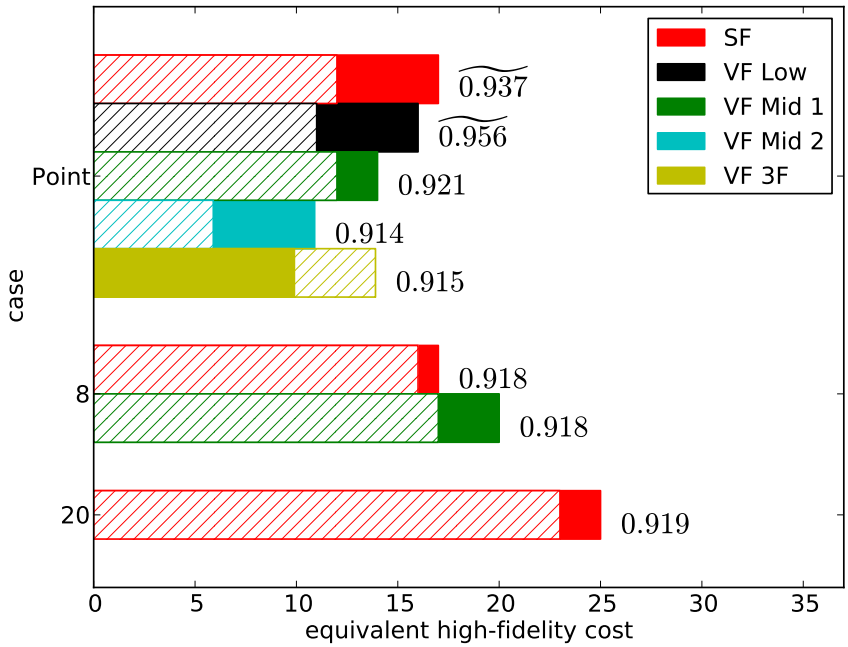


(a) Hover (threshold 0.916)

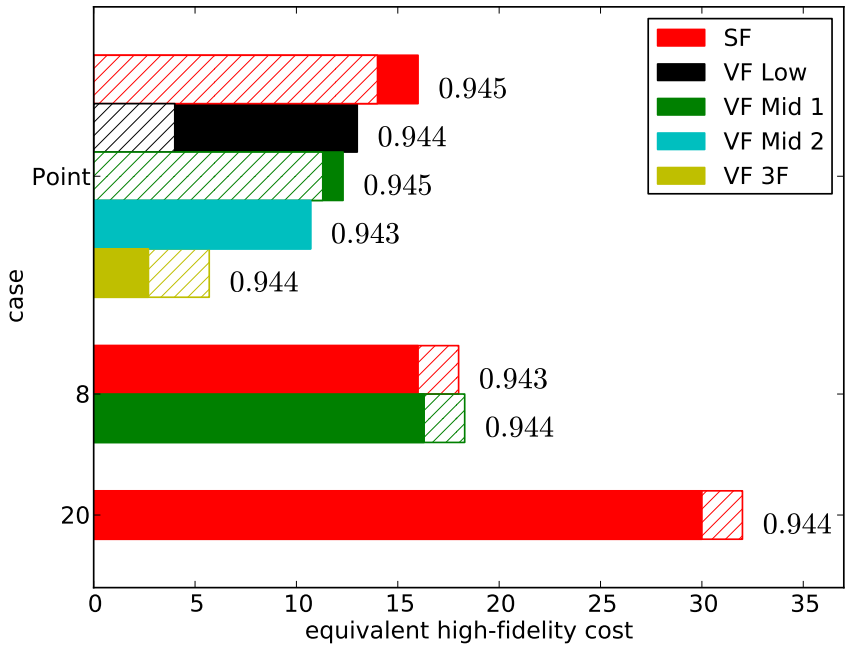


(b) Forward flight (threshold 0.944)

Figure 4.11: 7A single-objective optimization at **high-fidelity**: Convergence of single- (SF) and variable (VF)-fidelity. The ▽ represents the initial low-fidelity costs.



(a) Hover (threshold 0.916)



(b) Forward flight (threshold 0.944)

Figure 4.12: 7A single-objective optimization at **high-fidelity**: Cost to reach either threshold (filled area) or design confidence (DC) criterion (hatched area). The numbers are the goal function values upon reaching DC. SF = single fidelity, VF = variable fidelity. Results with a tilde (˜) highlight that the goal function value is above the 0.01 margin for the DC criterion.

timum is 2.7 for the **VF Point 3F** case and 10.7 for the other with the threshold set to 0.944, 0.001 worse than the best run. For acquiring the design confidence the cost is 5.7 and 10.7, respectively. The most cost efficient run on the single-fidelity side is the **SF Point** case with 16 for the optimum and 14 samples for the design confidence, which is considered valid here. Setting this into relation with the **VF Point 3F**, the cost savings are then 83.1% and 59.3%. Two factors are made responsible for this. First, the low-fidelity optimum is close enough to the high-fidelity optimum. This is extracted from Figure 4.11(b), where the initial points from the design of experiments are less than 0.003 in goal function value away from the optimum. Second, the better correlation among the methods in forward flight augments the high-fidelity surrogate well. The relatively little cost of the design confidence for the variable-fidelity case is linked to the benign behavior of the forward flight goal function for the high-fidelity. Looking back at Section 4.1.1, the landscape of the high-fidelity has a more gradual descent towards the minimum in comparison with the mid-fidelity. In return, this leads to larger spans of the radial basis functions (small θ), thus the influence of a single basis function is increased. The probability of improvement is reduced, or inversely design confidence is gained. It is pointed out that despite the high cost for reaching the defined threshold, the **VF Point Low** case reaches the design confidence the earliest. The starting point is the second best initial sample after the initial sample of **VF Point 3F case**. This indicates that the correlation between low- and high-fidelity is also very good, but getting the final precision simply takes longer than with the other methods. If the threshold would have been set according to the design confidence margin, 0.953, then **VF Point Low** case would be the fastest method to find the optimum with a cost of one high-fidelity sample.

The potential of the variable-fidelity approach is greater for this optimization at high-fidelity grade in contrast to the mid-fidelity gains. The factor in computational time for a high-fidelity simulation to a mid-fidelity simulation is about 35 for hover as well as forward flight. This means that the mid-fidelity cannot be as vastly sampled anymore as the low-fidelity. In hover the improved prediction of the mid-fidelity aligns a lot better with the one from the high-fidelity; therefore the variable-fidelity process is fully exploited. This is even better in forward flight, where the low- and mid-fidelity are almost equally good, and by using the low-fidelity as the trend function instead of the mid-fidelity, the costs are driven down even further.

Three important notes have to be made:

- Due to the higher cost of the mid-fidelity, an extensive surrogate based optimization is recommended at mid-fidelity beforehand. This is less costly than globally sampling the mid-fidelity.
- Triple-fidelity setups can, but do not have to be faster than dual-fidelity setups. They are only useful if the mid-fidelity is costly and the low-fidelity points the mid-fidelity into the right direction.
- The model error of the Kriging surrogate depends on the number of samples and the landscape of the function. Hilly or steep landscapes require more samples to reach design confidence than a simple flat landscape.

4.1.4 Multi-Objective Optimization

Trading-off between goal functions, here, hover and forward flight performance, is especially important for the design of helicopter rotor blades. As the balance between these two is very mission dependent, the weighted objective function only yields one specific design for one specific mission. Conducting the multi-objective optimization based upon the Pareto optimality criterion, many different designs can be identified and other advantages or disadvantages of certain geometries can be weighed aposteriori. Unless the Pareto front degenerates to a single configuration, which is usually not the case for helicopter rotors, the underlying surrogate model has to be accurate in a wide range of the parameter space. Not just a single region is of interest as is the case for the single-objective optimizations.

Mid-Fidelity Optimization

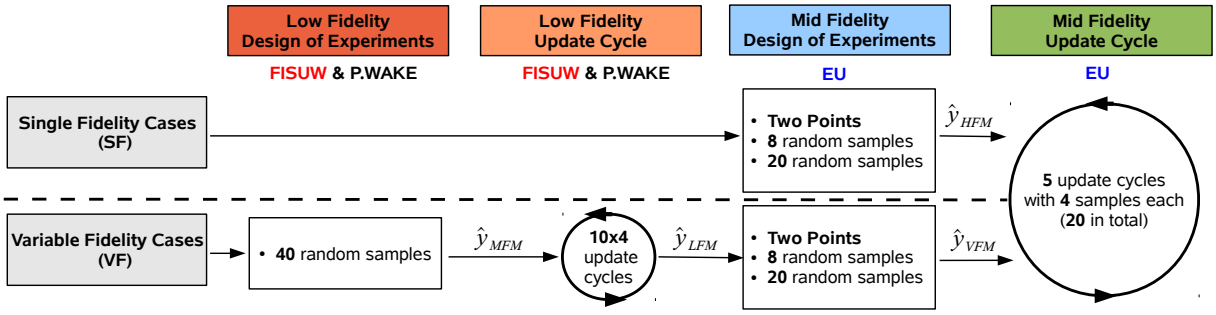


Figure 4.13: Test setup for the 7A mid-fidelity multi-objective tests.

The multi-objective optimization cases use the parameterization of the forward flight as the same sweeping restrictions apply, see Table 3.3 in Section 3.3. Therefore, the global hover optimum of the single-objective optimization cannot be found, but the local minimum becomes the best possible solution of the smaller design space. The effect of the randomization is not investigated. Thus, each hypercube is only generated once and then processed by either the single- or variable-fidelity optimization. The point samplings consist of either the low-fidelity minima from each flight condition for the variable-fidelity optimization, or the 7A and a 7AD similar blade for the single-fidelity optimization. The 7AD similar blade has the same proportions as the 7AD blade, however it is scaled with the thrust weighted chord length to match the 7A blade and the here utilized parameterization.

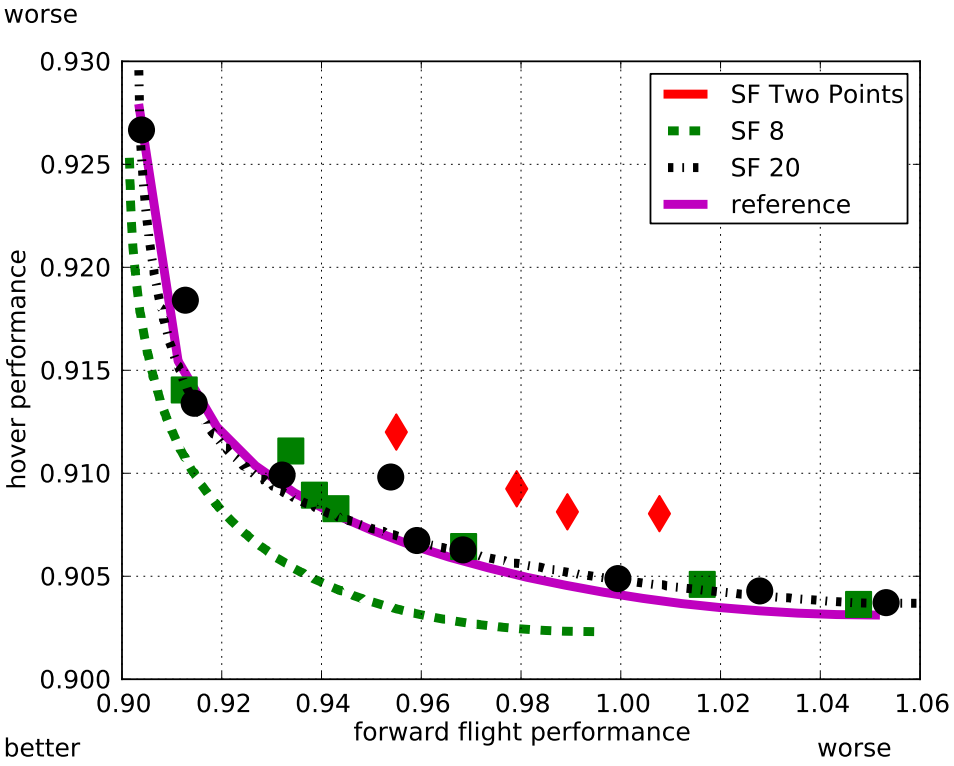
In Wilke [2] the optimizations are organized with a sequential sampling strategy, which proved the success of the multi-objective approach. In opposition to the sequential strategy, the parallel infill strategy explained in Section 2.4.2 is presented here. As it leads to similar results, the parallel infill is thus verified. In Figure 4.13, the test cases are explained. The variable-fidelity is started off a low-fidelity surrogate based optimization, which is initialized with 40 random samples with a combination of **FISUW**/**P.WAKE** for hover and forward flight. The update cycle for the low-fidelity is also performed in parallel with four samples per update cycle and ten update cycles, thus a total of 40 update samples is evaluated. Following this, three design of experiments are tried; **VF Two Points** selects the anchor points of the Pareto front, the optima from each individual goal function, to be sampled by the mid-fidelity (**EU**).

VF 8 and **VF 20** are simply sampling eight and 20 random samples. The same setup for the single-fidelity design of experiments is applied. Both modes of fidelity then perform five update cycles, where a maximum of four rotors is chosen to be evaluated in parallel according to the multi-objective optimization strategy from Section 2.4.2.

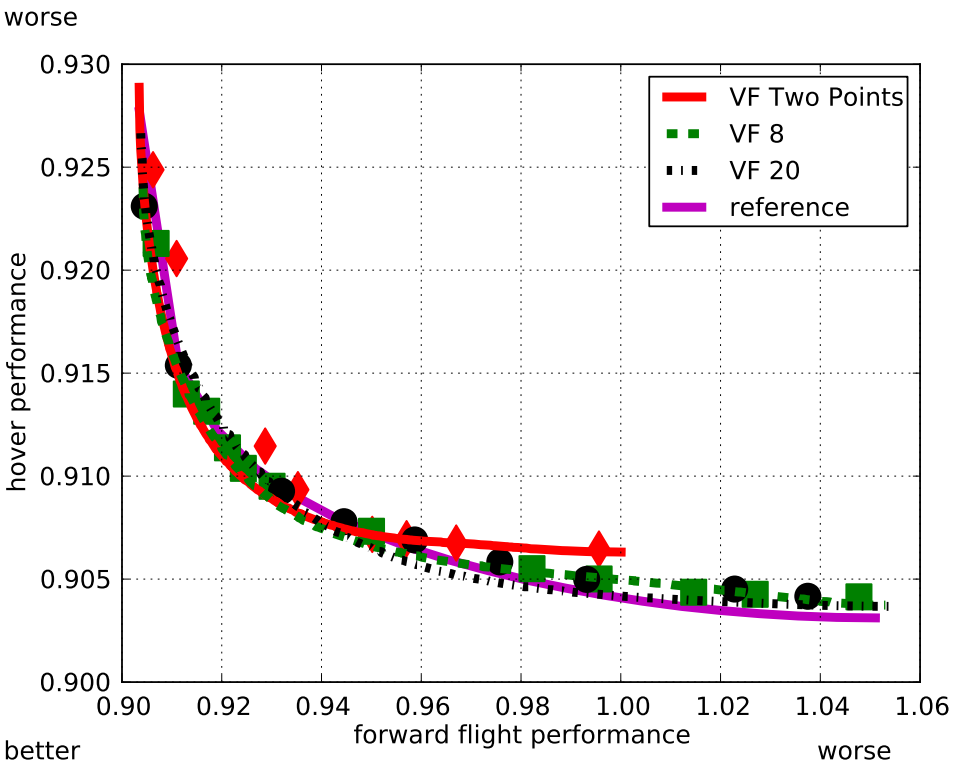
One difficulty lies in the benchmarking of multi-objective optimizations. As not a single optimal point exists, it is difficult to find a metric which marks the end of the optimization. The first metric is defined by the cost to reach both anchor points, the best hover and forward flight configuration. This still does not say anything about the intermediate designs. Therefore, the cost to reach the Pareto confidence, defined in Section 2.3.4 is compared as well. To further identify which method is better, the final Pareto optimal sets as determined from each method are compared against a reference solution visually.

The anchor points are known from the previously undertaken study of the performance landscapes in Section 4.1.1. Two of the four parameters are driven towards their limits, namely chord and sweep. Further exploring the design space in the surrogate models, it is observed that the intermediate designs also stay at the limits for these parameters, and therefore the optimization problem is simplified to a two dimensional problem to generate a reference solution. 81 samples are spawned in a full factorial design in the reduced design space of anhedral-twist to generate a very accurate surrogate model. In this surrogate model, the Pareto front is found by the same mechanism as described in Section 2.4.2 and poses the reference solution. A successful multi-objective optimization should be able to resemble this reference solution. On the one hand the shape of the front should match, and on the other hand the distribution of the according design variables should be close to the reference. This has to be true for the samples as well as the surrogate version of the Pareto front. A faster method should then have a more advanced Pareto front, than a slower method, when the same amount of resources is granted.

In Figure 4.14(a) and Figure 4.14(b), the surrogated Pareto fronts are plotted as dashed lines (--), the reference solution (—) as a magenta line, and the actual samples as symbols. The symbols for **Two Points** are red diamonds (◊), for **8 random** are green squares (■), and for **20 random** black dots (•). For the single-fidelity it is observed that the solutions strongly vary with the size of the design of experiments. The approximated front for the **SF 20** case, black line, is close to the **true samples**, black dots, as well as the reference solution. This shows that the single-fidelity is able to obtain a valid solution for the Pareto front. Also, when looking to the potential parameter combinations in Figure 4.15(a), the black **SF 20** scatter is in good standing with the reference solution. This becomes worse for decreasing the number of samples in the design of experiments. Particularly the **SF Two Points** case, red diamonds, is not close. The few samples are scattered above the reference front with the approximate Pareto frontier strongly underestimating both goal functions. The surrogated front is outside the plot on the bottom left. For the parameter combinations, a distinct area around anhedral 0 and twist -14^o is noticed. This area belongs to an artificial second minimum in the surrogated hover function, which is not part of the true Pareto front. Due to the inaccurate surrogate model of the **SF Two Points** case, these artifacts are visible on the parameter plot. The **SF 8** case demonstrates well that more samples aid the process and obtains an intermediate solution between **SF Two Points** and **SF 20** case. It places most samples close to or on the reference solution, but the surrogate model still lacks accuracy when analyzing the parameter distribution, displayed in Figure 4.15(a).

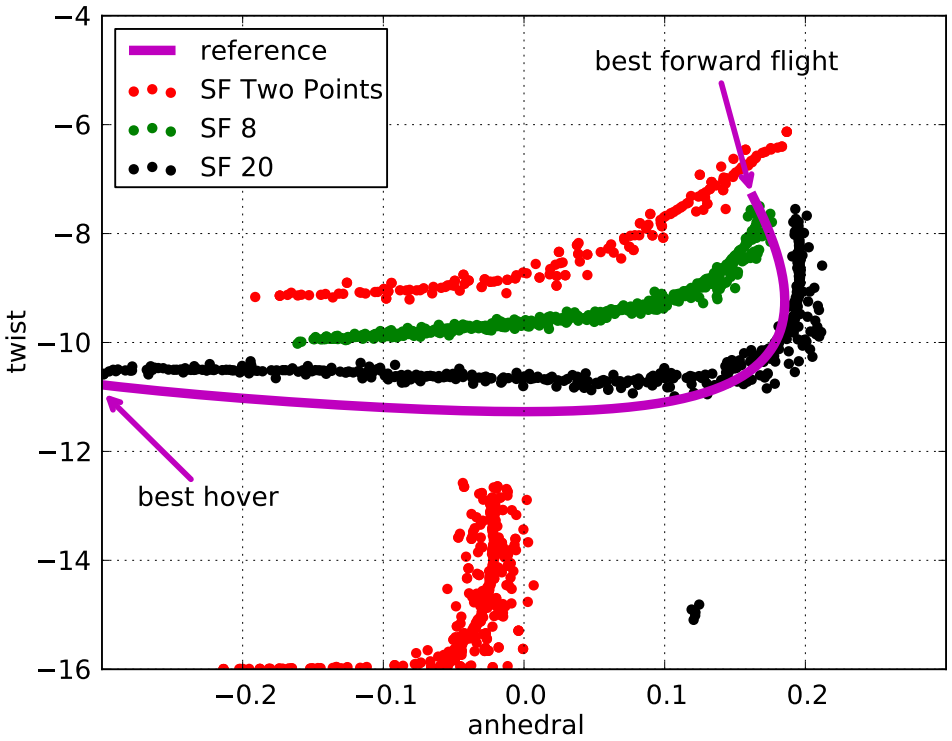


(a) Single-fidelity

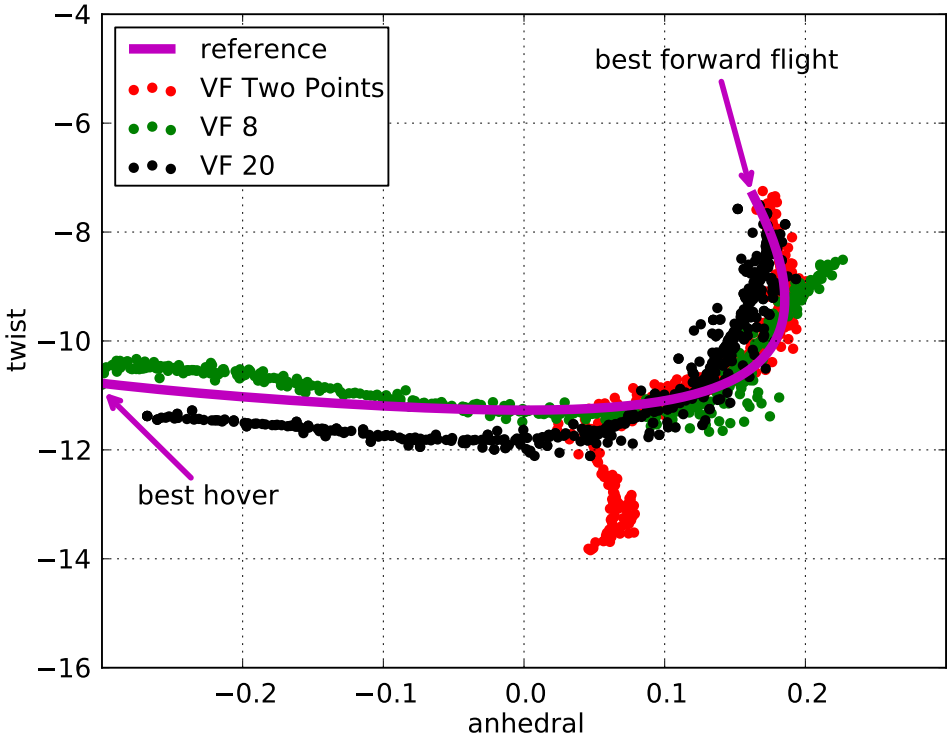


(b) Variable-fidelity

Figure 4.14: Pareto fronts at mid-fidelity level for the 7A rotor optimization. Points are simulated points, lines correspond to the Pareto fronts within the respective surrogate models.



(a) Single-fidelity



(b) Variable-fidelity

Figure 4.15: Pareto optimal parameters at mid-fidelity for the 7A rotor optimization. Scatter points correspond to the Pareto fronts of the surrogate models in Figure 4.14.

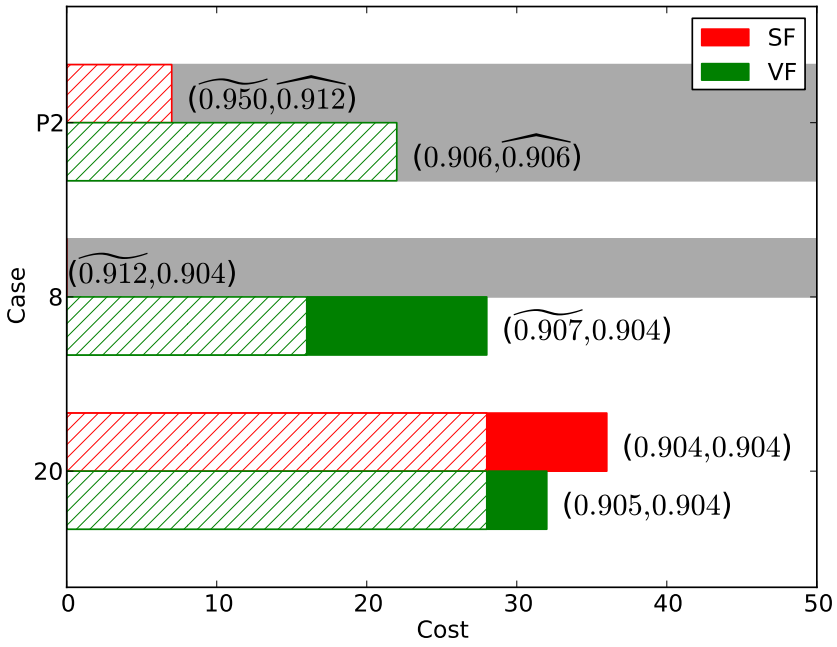


Figure 4.16: The cost of each **7A mid-fidelity multi-objective optimization** for reaching the threshold goal functions (GF) of 0.907 for forward flight (FF) and 0.905 for hover (HV), filled area, or having two consecutive iterations with 90% Pareto confidence (PC), hatched area. The numbers denote the current best goal functions upon reaching PC. Results with a tilde (˜) highlight that the goal function value is above the 0.01 margin for the PC criterion. Gray bars indicate that an anchor point is not found.

The tendency that more points improve the resolved Pareto fronts is also observed for the variable-fidelity approach, but is less critical than the single-fidelity approach. In Figure 4.14(b), the point sampling, **VF Two Points**, gets very close to the reference solution, but misses the hover minimum as it is also driven towards the virtual second hover minimum just like the **SF Two Points** case. This is also highlighted by the parameter distribution in Figure 4.15(b). The **VF 8** and **VF 20** cases are almost of equal quality. They both place samples along the line of the reference solution in Figure 4.14(b), and at the same time accomplish a good agreement with the parameter combination of the reference solution in Figure 4.15(b), without getting stuck in the local minimum.

The findings about the accuracy are also reflected in terms of reaching the Pareto confidence as well as the recovering the anchor points. In Figure 4.16 the individual costs are pictured for each case. The threshold for the anchor points is chosen to be 0.907 for forward flight and 0.905 for hover. These are well below the allowed margin of 0.01 for the Pareto confidence, yet is chosen to have a more competitive comparison. **SF Two Points**, **SF 8** and **VF Two Points** therefore fail to find both anchor points, and are thus considered incomplete optimizations on academic terms. For engineering purposes, the local minimum might be just as suitable as the global one, since the goal function values only differ by 0.001, which is in the allowed Pareto confidence margin.

On the variable-fidelity side the **VF 8** case is the most efficient case, whereas for the single-fidelity the **SF 20** case is the only successful one. The reduction between these two cases measured in the cost for finding the anchor points is 12.5% while the Pareto

confidence is reached quicker and 42.9% resources are saved. In contrast to the single-objective optimization, this means less benefit. This is accredited to the not well matching low-fidelity trend for the hover flight condition. It requires more corrections for the multi-objective scenario since a more globally accurate surrogate model is required than for the single-objective scenario. This leads to the conclusion that the multi-objective scenario suffers from the "weakest link", the goal function which is enhanced the least by the variable-fidelity approach.

Summing up the findings, the following two statements are made:

- The proposed hybrid multi-objective methodology (Section 2.4.2) works well in parallel and is able to find valid Pareto fronts.
- The variable-fidelity multi-objective approach suffers from the weakest link. The reduced correlation of the low- to the mid-fidelity in hover is the bottle-neck for the multi-objective optimizations.

High-Fidelity Optimization

Just as with the single-objective optimization for high-fidelity, the focus for the high-fidelity multi-objective optimization is more laid onto the mix of the various trends given from low- and mid-fidelity methods, rather than finding the optimal size for the design of experiments. In Figure 4.17 the combinations of low-, mid- and high-fidelity samples are displayed. Due to the costly nature of the high-fidelity samples, only one single-fidelity run is compared with three variable-fidelity runs. One hypercube containing eight random high-fidelity samples is generated for the **SF 8** and **VF 8 3F** case. In the **VF Two Points 3F** and **VF 8 3F** cases, a triple-fidelity setup is employed. From the mid-fidelity optimization, the surrogate models of the **VF 20** case are recycled. One reason to do so is found by the single-objective optimization at high-fidelity level from Section 4.1.3. The triple-fidelity setups are among the most efficient techniques, yet at the same time pre-refined regions also improve the speed. For the **VF Two Points 3F** case, the design of experiments consists of the best rotors for each flight condition, hover and forward flight, at mid-fidelity. As the major resource consumption is given from the forward flight simulation, the **VF Two Points Mix** applies the surrogate models for forward flight from the low-fidelity prescribed wake model (**P.WAKE**) and for hover the model of the variable mid-fidelity (**FISUW** and **EU**) multi-objective optimization from the **VF 20 case**. Thus, a dual-fidelity setup for forward flight and a triple-fidelity approach for hover are employed. This is the least resource intense setup, when looking at the initial as well as overall granted resources. A setup with both trends given from the low-fidelity is not investigated; from the single-objective high-fidelity hover optimization it is concluded that the hover trend of the low-fidelity actually slows down the high-fidelity optimization. For all run setups, four infill samples are computed during five update cycles in the same manner as it is done for the mid-fidelity optimizations.

A comparison of all four obtained Pareto Fronts from each optimization is depicted in Figure 4.18(a) with the parameters anhedral, sweep, and twist in Figure 4.18(b), Figure 4.18(c) and Figure 4.18(d), respectively. The continuous magenta line denotes the reference solution (—), which is obtained from merging all data points from all runs together and keeping only the ones not dominated within this set. For each parameter a regressive Kriging model is created in dependency of the forward flight performance

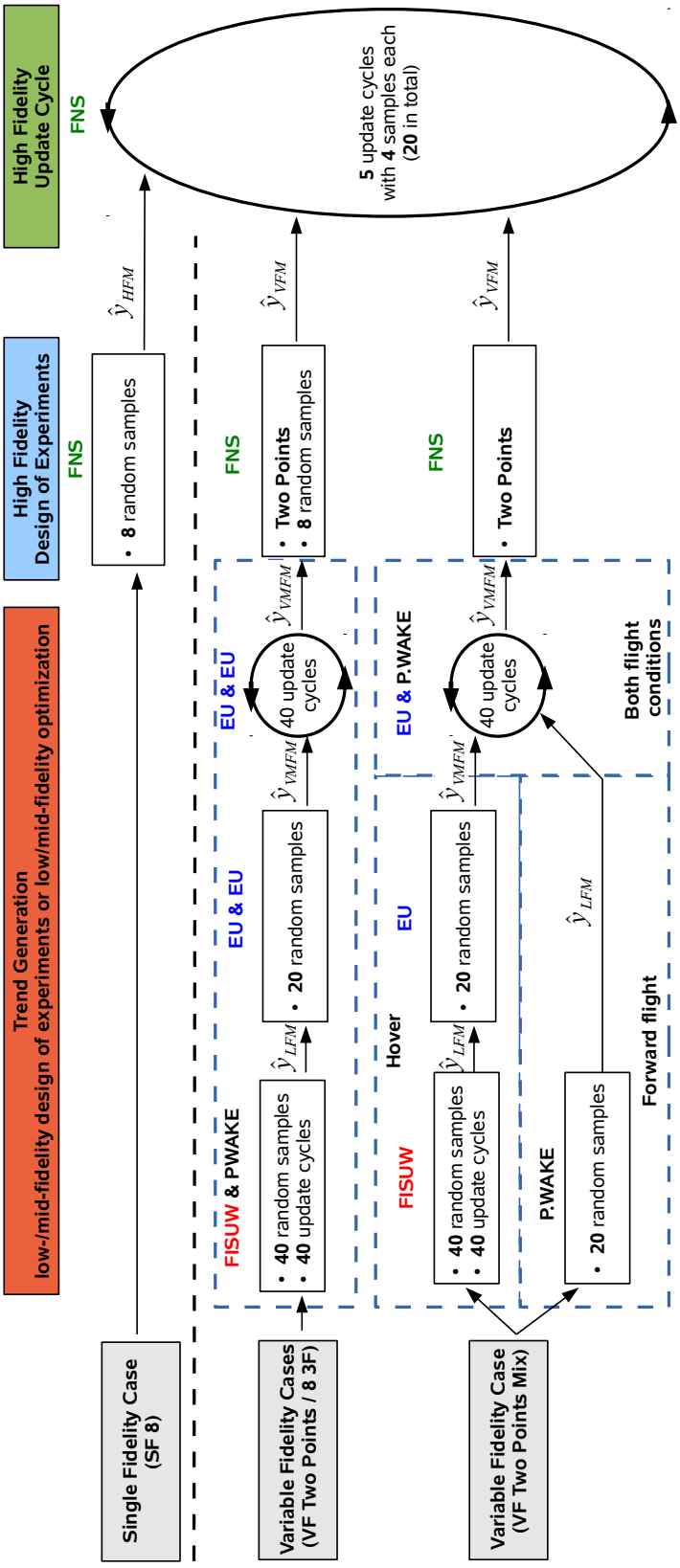
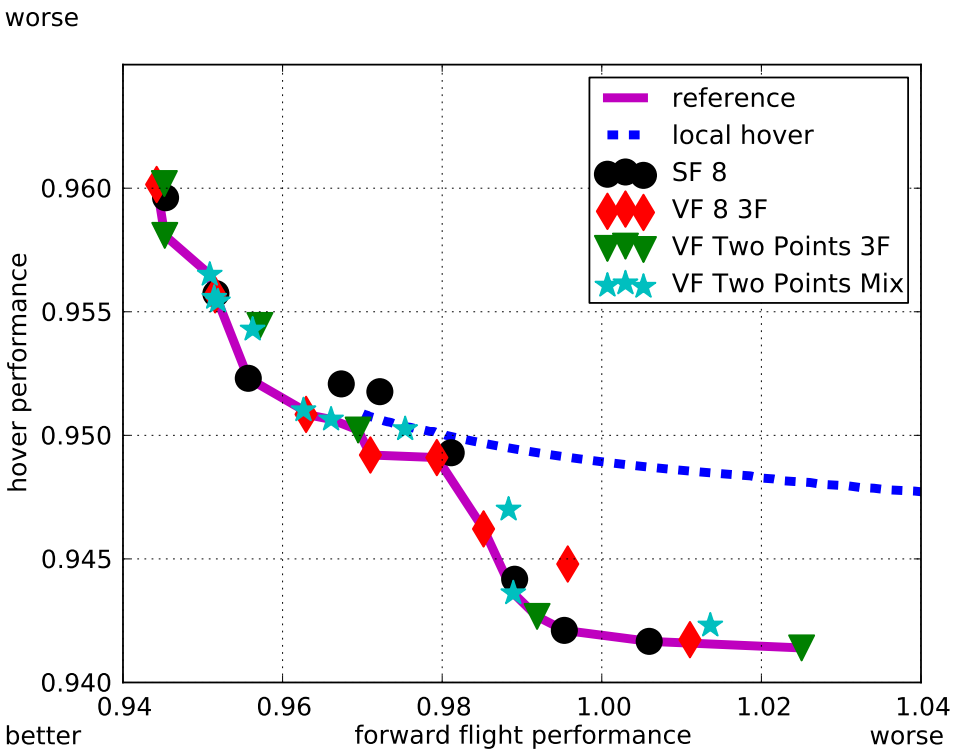
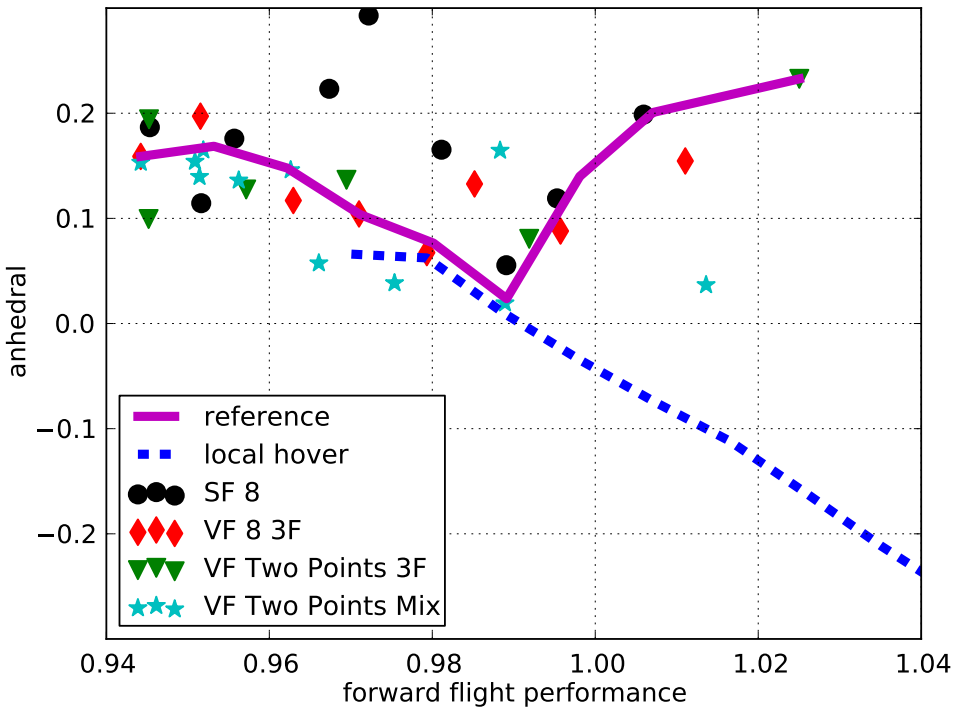


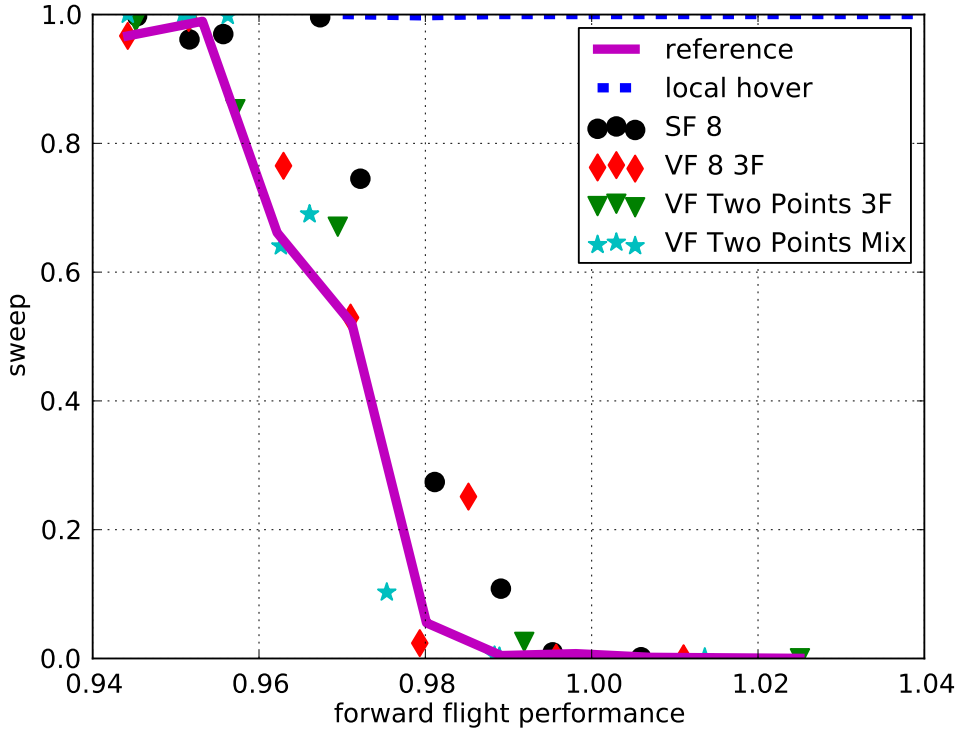
Figure 4.17: Test setup for the 7A high-fidelity multi-objective tests. Dashed blue boxes represent the re-use of the mid-fidelity surrogate models, see Figure 4.13



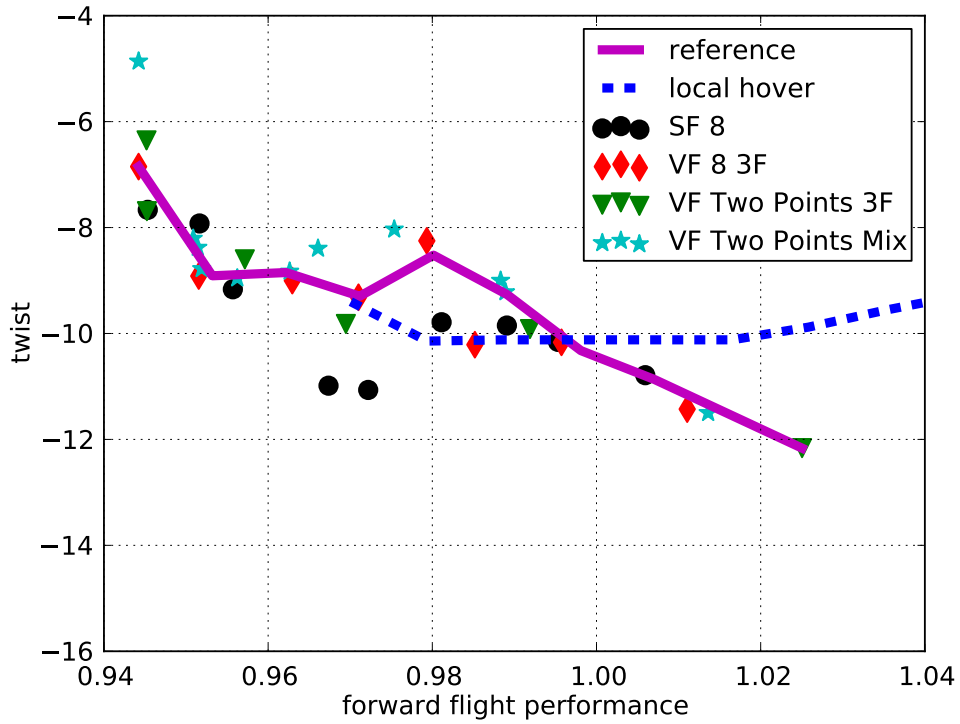
(a) Pareto front



(b) anhedral



(c) sweep



(d) twist

Figure 4.18: Comparison of high-fidelity multi-objective results for the 7A rotor after 20 update cycles confidence. Magenta shows the reference solution (—) abstracted from all evaluated samples and is slightly smoothed on the parameter plots for better visibility. The dashed blue line represents an alternative solution (--) going to the local hover minimum.

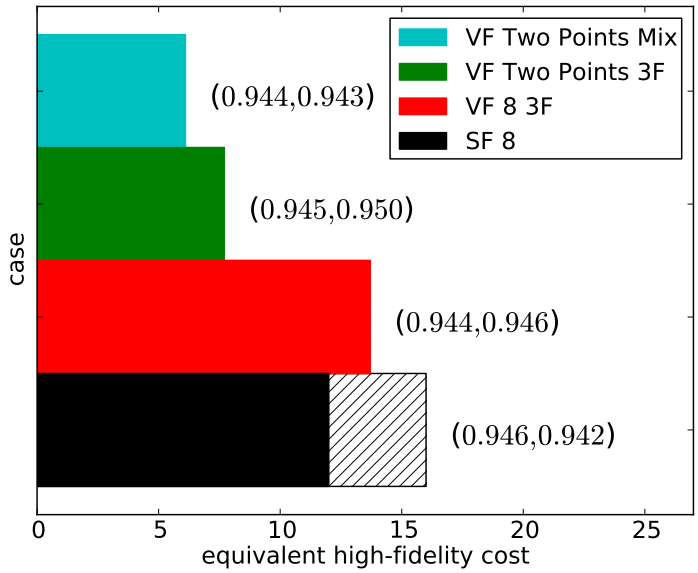


Figure 4.19: The cost of each **7A high-fidelity multi-objective optimization** for reaching the threshold goal functions (GF) of 0.954 for forward flight (FF) and 0.951 for hover (HV), filled area, or the first iteration with 90% Pareto confidence (PC), hatched area. The numbers denote the current best goal functions upon reaching PC.

to visualize their behavior. The symbols mark the discrete results of each test case; the black dots are for the **SF 8 (•)** case, the red diamonds for the **VF 8 3F case (⋄)**, the inverted green triangles for the **VF Two Points 3F (▽)** case and the cyan stars for **VF Two Points Mix (⋆)** case.

At the point $(y_{ff}, y_{hv}) = (0.98, 0.948)$ a kink is recognized in the reference Pareto front, also observed in the parameter sets. It is best seen in the sweep plot, Figure 4.18(c), but also in the anhedral plot Figure 4.18(b). A thesis is stated, which claims that this kink occurs, as the Pareto front first goes from the forward flight optimum towards the local hover minimum, but roughly at the point $(y_{ff}, y_{hv}) = (0.98, 0.948)$, the Pareto front shifts towards the global hover optimum. This is further examined by starting a local searcher at the position of the kink and then letting it optimize the hover goal function. This is done in the surrogate models created from all high-fidelity samples computed for the multi-objective test. All the points the local searcher visited are recorded and make up the dashed blue line representing this alternative solution (--). Checking the obtained minimum from the local searcher proves this thesis as it correlates with the one found from the landscape investigation in Section 4.1.1

Going into the evaluation of the different Pareto fronts, the first observation is that **VF Two Points Mix** contributes six points to the reference Pareto front, **VF 8 3F** five, **VF Two Points 3F** four and **SF 8** three. However, this metric alone does not reflect the whole truth. The **VF Two Points Mix** case places many points along the reference Pareto front, but does not capture the exact anchor points, Figure 4.18(a). Opposing this, the **VF Two Points 3F** provides the hover anchor point and is only mildly superseded by the **VF 8 3F** case in the forward flight anchor point. Comparing this anchor point with the single-objective results of the high-fidelity optimization, a relative difference of 0.063% is noted. The kink is resolved by all methods judging from the sample density in the area of all methods. It is pointed out in Figure 4.18(a) and Figure 4.18(b)

that the **SF 8** case has some outliers around $y_{ff} = 0.97$, which is found in the anhedral plot of Figure 4.18(b). These outliers are a result of a yet inadequate resolution of this area in the surrogates of this case.

This graphical interpretation of the quality of the individually resolved Pareto fronts insufficiently distinguishes the merits of each process. Thus, the cost for reaching either the anchor points within a specific threshold and the number of resources required to obtain the Pareto confidence for the first time are charted in Figure 4.19. The threshold is set to 0.954 for forward flight and 0.951 for hover, 0.01 above the respective anchor point values. The numbers on the right side of the bars are the best goal functions values sampled so far upon reaching the Pareto confidence. For the variable-fidelity cases, reaching the threshold and the Pareto confidence coincides to the same number of resources, while for the single-fidelity case, the Pareto confidence is reached one update cycle later. The most efficient case on the variable-fidelity side is the **VF Two Points Mix** case. Comparing this with the single-fidelity case, the best variable-fidelity setup requires 50.8% of the resources to reach the threshold and about 38.1% for the Pareto confidence.

Two conclusions are drawn from the multi-objective high-fidelity tests:

- The setup, where the forward flight low-fidelity is coupled with a mid-fidelity hover method proves to be the most efficient scenario. Reason for this is that the mid-fidelity hover function is relatively cheap and the forward flight low-fidelity is almost for free. This reduces the initial costs significantly in comparison with the other variable-fidelity setups.

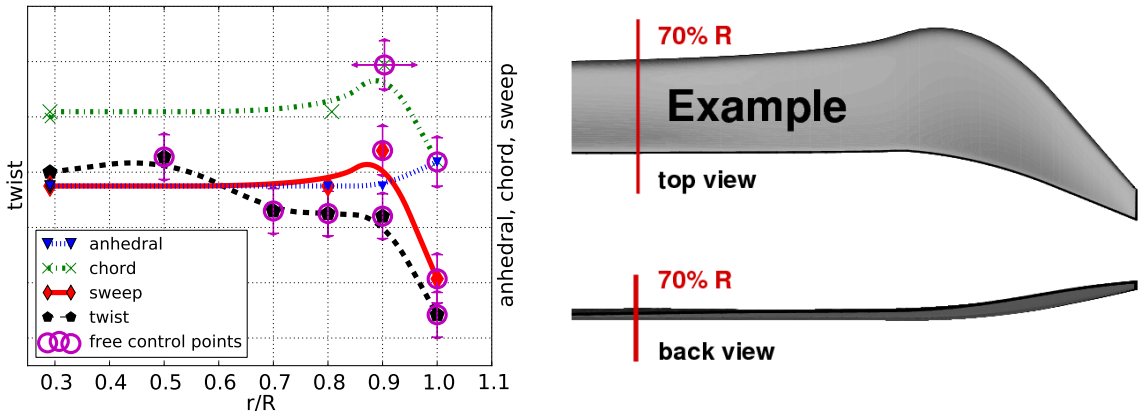
- The major benefit from the investigated variable-fidelity cases comes from the ability to employ smaller high-fidelity design of experiments to capture the goal function landscapes seen in Figure 4.19.

4.2 Industrial Scale Rotor Optimization with Ten Parameters

The baseline rotor for this test is taken from the European Joint Technology Initiative Clean Sky Green Rotorcraft Research (GRC) Program [150]. The rotor has a tip radius of 5.5 m, 5 blades with a parabolic blade tip similar to the 7AD configuration and a linear twist throughout the airfoiled sections. The optimization itself is taken a step further from the 7A optimization, now a structural constraint is included. The absolute peak torsional moment at the blade root at about the location of the pitch link rods is constrained to the value of the baseline blade in each individual flight condition. The two considered flight conditions are again forward flight and hover. The advance ratio in forward flight is lower in comparison to the 7A rotor and corresponds to $\mu = 0.33$. The thrust in hover is about one third higher than in forward flight. For the multi-objective optimization, both structural constraints from hover as well as forward flight are included. As the baseline blade is proprietary, the figures and plots are dimensionless and geometric properties are only relatively quantified. They are also of minor importance for the scientific statement to be made. The major purpose of this test is to raise the complexity of the optimization and quantify the advantage or disadvantage of the variable- over single-fidelity approach. The focus is laid less onto the surrogate-based optimization, but on the physical phenomena to be modelled that lead to the success or failure of the variable-fidelity approach.

4.2.1 Parameterization

As the reference rotor system is already an advanced system, the number of considered parameters is increased. The here presented parameterization is the final outcome after preliminary trials and the work done by Heilers [137] to ensure that upon performing the design of experiments not more than 20 % of the simulations returned unflyable configurations as no successful trim solution could be obtained. Though Heilers has up to 60 % untrimmable rotors and the given (single-fidelity) framework including the crashmap (Section 2.5.4) can deal with it, the ratio of cost to gain is strongly reduced.



(a) Plot of parameterization

(b) Example rotor blade of parameterization

Figure 4.20: Parameterization and example blade of GRC rotor optimization

The parameterization is based upon second order Non-Rational Uniform B-Splines

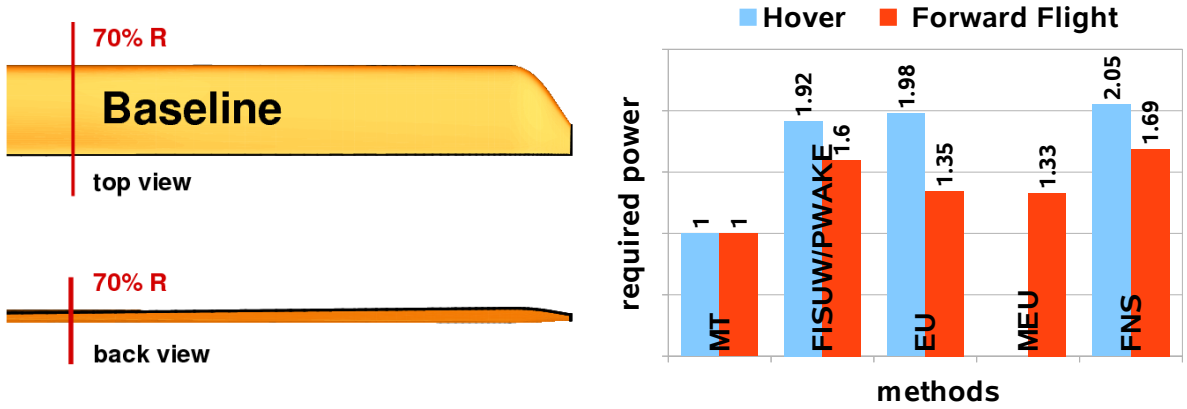
(NURBS [151]), which allow the creation of a continuous and smooth line for each parameter. The downside is that the control points do not directly lie on the obtained curve, which is the price for smooth distributions. For the anhedral, chord, and twist parameters the free control points of the NURBS can be moved in the ordinate direction by the optimizer, while the control point for chord length can also be moved in the radial direction. For the chord length, it is decided that it is most beneficial not to vary the outmost blade tip chord length, which is already decreased for the baseline blade, but utilize an inboard parameter to control the length and another parameter to control the position of this length. Two sweep parameters are employed; one placed at the very tip and one slightly more inboard. The inboard one can only move forward, while the outboard one can only move aft. The optimizer then has the option to counter a torsional moment caused by the backward sweep on the outboard station with a forward sweep at the inner station. Only one anhedral parameter is employed. Heilers [137] demonstrates that in conjunction with a twist optimization, the effect of a higher dimensional parameterization of the anhedral brings negligible benefit. As twist is the most dominant driver, it is discretized with five parameters to give the optimizer enough degrees of freedom. This is also reasonable, as this helps the optimizer to consider sweep and twist decoupled from each other. Otherwise with a small set of parameters the optimizer carries out unwanted aero-elastic tailoring to match the local twist through sweep as Imiela [125] finds out in his work. For clarity, in Figure 4.20(a) the planform parameters are plotted on the left, while the blade tip of these exemplary parameters is plotted on the right in Figure 4.20(b).

Compared to the simulations of the 7A rotor, the RANS meshes now include the blade tab and in forward flight the background mesh is refined. More details on the discretization of the individual methods along with a grid convergence study on the RANS meshes are found in Section A.2.3 and Section A.2.4.

So far, only the planform is changed in the aerodynamic solvers. Now an adaptation of the structural properties is included in the HOST beam model. It is based on the findings by Stanger [152] and Stanger et al. [153], who compute a difference of up to 2% in the results for required rotor power between adjusted and unchanged structural properties for the same blade. Thus, the stiffnesses are adjusted for each blade as well as the offset of the blade in case the sweeping or anhedral is enlarged in relation to the baseline rotor with the proposed strategy by Stanger. Additional notes concerning this approach are made in Section A.2.5.

4.2.2 Preparatory Fidelity Study

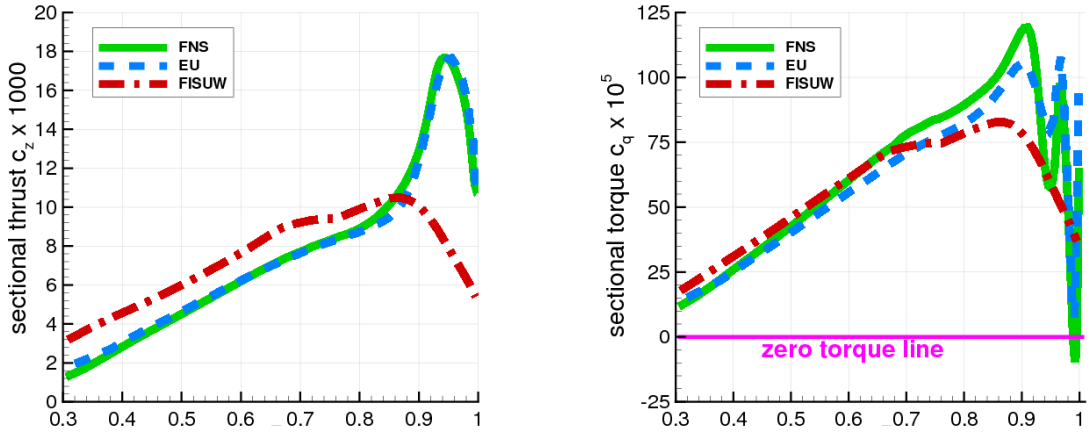
A fidelity convergence study is undertaken for the new rotor system examined in an abbreviated version to the fidelity study in Section 3.3. First the baseline blade and its different representation by the various methods is presented, before the blades from the optimizations at different fidelity levels are investigated. The benchmark of the optimizations themselves is then reviewed in Section 4.2.4 to Section 4.2.5. An additional method is added here, the **MEU** method is a multi-bladed Euler simulation. It consists of the same blade grid as the **EU** method with a coarsened version of the **FNS** background grid. A sensitivity study of the parameters is not undertaken here as the ten dimensional design space has too many interdependencies in the parameters and thus statements only at specific locations in this design space can be made without the possibility of generalization. Understanding the physics behind the different blades



(a) Sketch of blade

(b) Predicted required power (normed with the momentum theory results)

Figure 4.21: Sketch of the GRC baseline blade along with required powers



(a) Thrust

(b) Torque

Figure 4.22: Sectional loads of GRC blade with variable-fidelity methods in hover

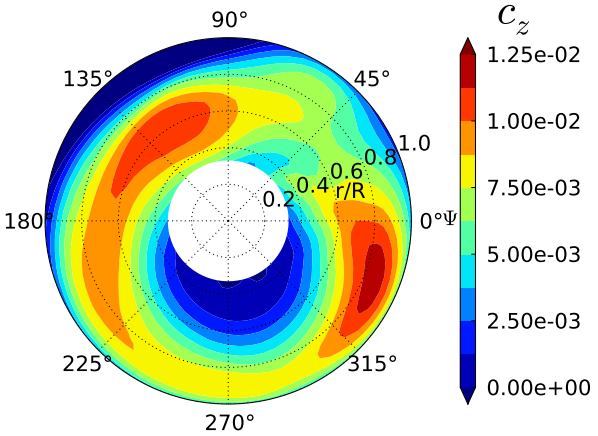
obtained is easier to grasp, thus the optimized blades are presented along with their driving physical effects.

The baseline blade is sketched in Figure 4.21(a) having a similar design to the 7AD blade. The predicted required power consumption for the selected fidelities is charted in Figure 4.21(b). They are normed with the solution of the momentum theory. For hover, the results are almost equal, which is surprising for the **EU** method as it neglects viscosity. This is identified to be the effect of the coarse mesh, which increases the numerical viscosity and thus raises the power requirements. In forward flight the ratio of the inviscid to the viscous computation is similarly to the 7A rotor simulations. The required power computed by the **EU** method is 20.1% and 21.4% for the **MEU** method less than for the **FNS** method. This is roughly the reported magnitude for the viscous torque of the **FNS** simulation. The **P.WAKE** method matches better with the **FNS** solution than it does for the 7A rotor.

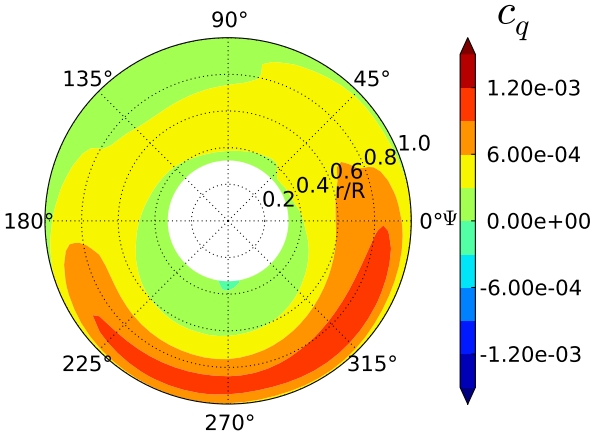
Looking purely at the single value of required power can be misleading and thus the load distributions are also briefly investigated here for the baseline blade. The sectional thrust distribution is plotted on the left of Figure 4.22, while on the right side the

torque distribution is plotted. While the **FISUW** method is out-of-line for the thrust as well as the torque distribution, the **EU** method follows the thrust distribution of the **FNS** method very well. The kink at around r/R = 80% is slightly relaxed. This is argued with the coarse mesh as this kink reflects the decreasing influence of the first tip vortex. The vortex core is wider on the coarse mesh and smooths the thrust distribution. The torque distribution is also very similar between these two methods. The difference from the employed viscosity is visible as up to r/R = 90 % the **FNS** method predicts an additional linearly growing torque. The wiggles at the end are a result of the interaction of the induced tip-vortex and the boundary layer. The wiggles are initiated at the beginning of the blade sweep at around r/R = 95 %, being in the upwash region of the previous vortex, which hits the blade approximately around 90%.

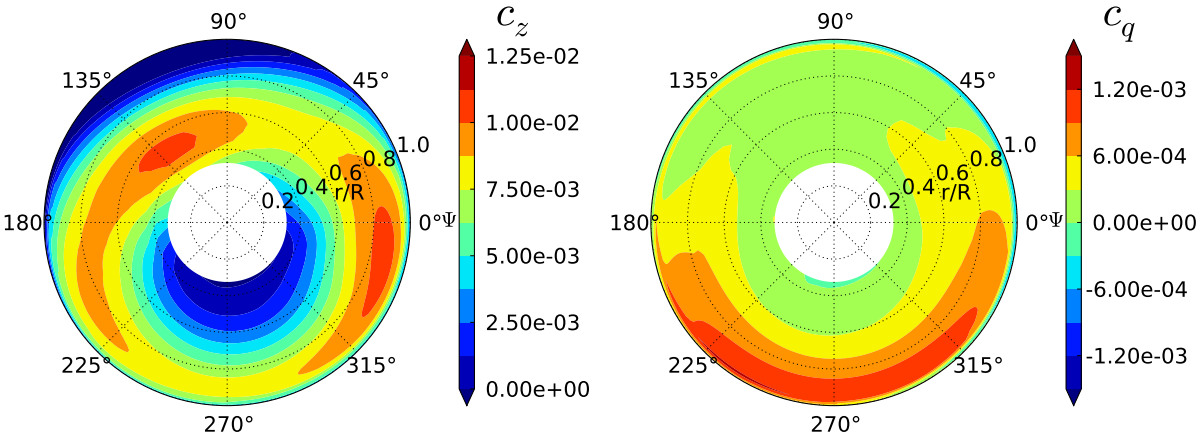
In forward flight, the better agreement among the methods is found in the thrust and torque plots in Figure 4.23. On a first glance the thrust distributions look very alike, but small important differences are noticed. The **P.WAKE**, **MEU** and **FNS** method predict the lowest thrust region on the advancing side, $\psi = 75° \ldots 180°$, on the outer blade identically, while the **EU** method predicts the area too early at $\psi = 45° \ldots 180°$. At the same time, the thrust peak is shifted slightly inboards at this location. Both of these differences are on the account of the discarded tip vortex in the **EU** method, as the elastic deformations are very similar for all results excluding viscosity from the argument. The most distinguishable difference between the **P.WAKE** and **FNS** method is the extended thrust peak at $\psi = 225° \ldots 270°$. It exists in the simulation of the **MEU** and the **FNS** methods, which is a local effect of the trailing vortex being differently resolved in the **P.WAKE** method. A similar tendency of the **P.WAKE** and **FNS** method is noted in the torque plots. The differences are that **FNS** generally predicts more torque and a small half-ring at r/R = 95% is found on the advancing side as well as an overall larger peak on the retreating side between $\psi = 225 \ldots 330$. Also, the inboard area is slightly rotated. The half-ring is the effect of the drag divergence as the advancing side operates in a transonic regime. This effect is also modeled by the **EU** method. The phase shift of the torque peak for the **EU** method on the retreating side arises from the lack of friction as well as the missing vortex representation. This phase shift is corrected for the **MEU** method, yet similar to the **EU** method the overall torque level is lower.



(a) Sectional lift for **P.WAKE** method

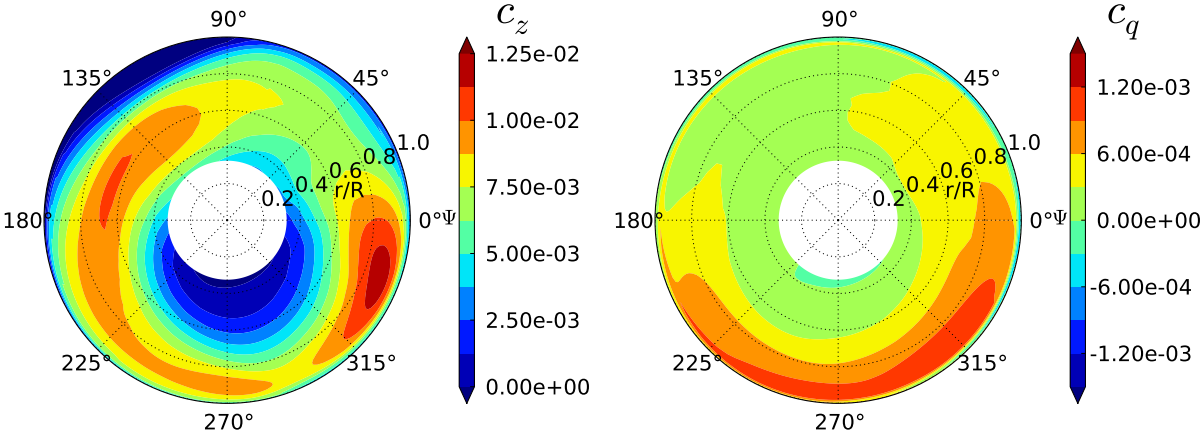


(b) Sectional torque for **P.WAKE** method



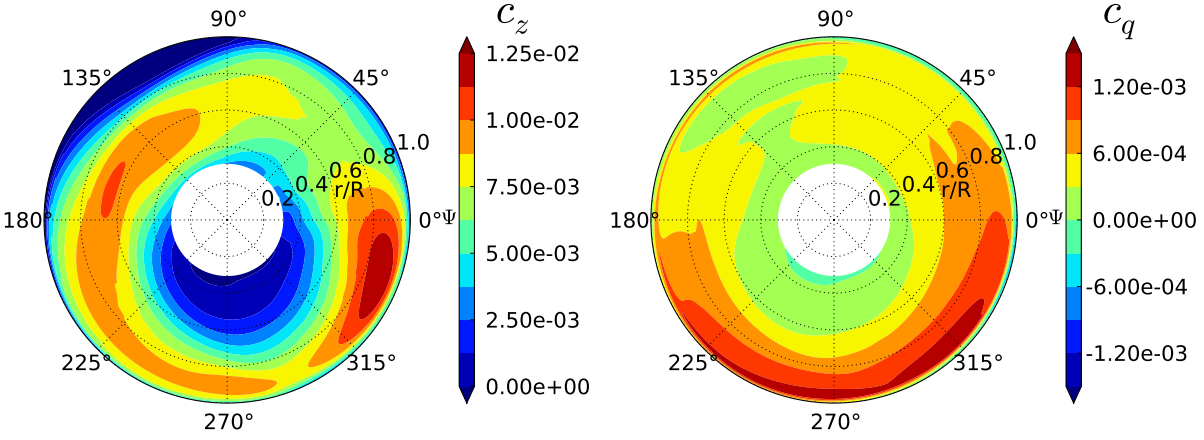
(c) Sectional lift for **EU** method

(d) Sectional torque for **EU** method



(e) Sectional lift for **MEU** method

(f) Sectional torque for **MEU** method



(g) Sectional lift for **FNS** method

(h) Sectional torque for **FNS** method

Figure 4.23: Comparison of loads among the various methods for the GRC blade in forward flight.

Hover Optimized Blades

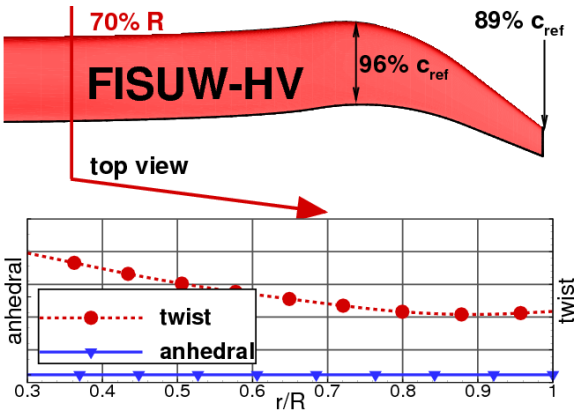
Starting with the hover optimizations, the best rotors for the low-, mid- and high-fidelity approaches are depicted in Figure 4.24(a), Figure 4.24(c) and Figure 4.24(e) with their respective difference of the loads in Figure 4.24(b), Figure 4.24(d) and Figure 4.24(f). Table 4.1 summarizes the improvements of these blades accomplished in reference with the baseline blade. The sketched blades have a clear visual difference, with the low-fidelity blade being furthest away from the high-fidelity blade. The major difference between the mid- and high-fidelity blades is the chord length at the blade tip. While the obtained low-fidelity blade looks more like a forward flight configuration, due to its moderate sweep and decreased chord length without any anhedral, the Euler blade features very high twist gradients towards the tip as does the high-fidelity blade.

Method	power reduction		pitch link loads reduction	
	predicted	checked	predicted	checked
FISUW	-2.0%	+10.8%	-8.5%	+13.3%
EU	-11.2%	-5.9%	-0.3%	+23.4%
FNS	-6.5%		-2.4%	

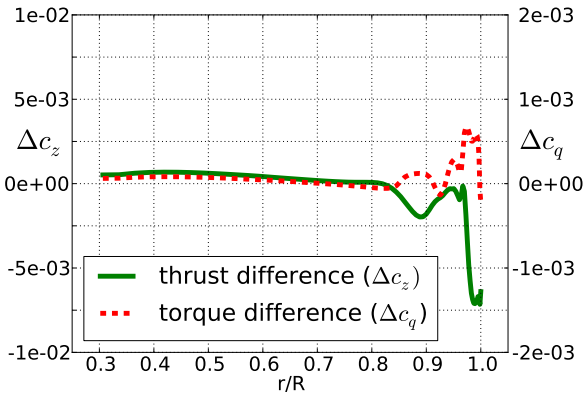
Table 4.1: Power reduction of blades of individual **hover** optimizations. The predicted reduction is given at the individual fidelity, the checked reduction is the result when re-computed at high-fidelity (**FNS**).

The low-fidelity blade is tapered earlier than the baseline blade and is swept slightly stronger. It does not have an an- or dihedral and shows an almost hyperbolical twist distribution. The latter is the desired outcome for the ideal blade when the blade element theory is coupled with the momentum theory. However, for a real blade this twist distribution is not wanted as the tip vortex is neglected. Looking at Figure 4.24(b), a raise in thrust in comparison with the baseline blade is noticed at around r/R = 80% , followed by a strong decrease at r/R = 97% accompanied with a raise of torque at r/R = 95 %. This is attributed with the little twist gradient at the tip leading to a small separation there. The tip vortex is released and hits the following blade at around r/R = 90%. The section more inboard of this location experiences an additional downwash, while the other side sees an additional upwash. This upwash part increases the angle-of-attack to a point, where a small separation occurs. It becomes clear that the low-fidelity method is not suited for higher dimensional hover optimizations, as it cannot model the tip vortex properly and thus the stalling part of this blade is not resolved by **FISUW**.

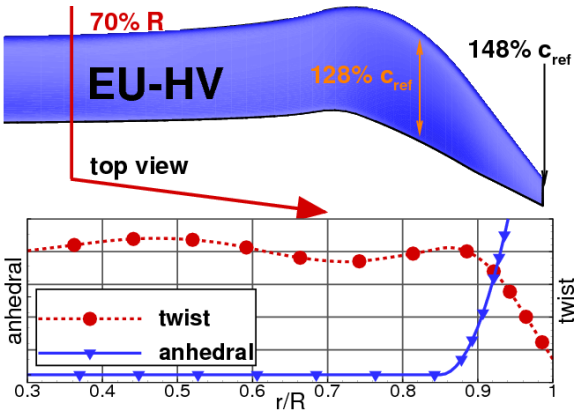
The mid-fidelity blade does not have this stalling region and features an enlarged chord length in the tip region, a greater forward/backward sweep tendency and an anhedral larger than for the low-fidelity blade. As the mid-fidelity method resolves the tip vortex, it is able to exploit this in the optimization. The last 5% of the blade actually regain energy from the tip vortex by twisting the blade in such a way that the resulting force vector points forwards. The flow then comes in at a relatively low angle of attack countering the upwash from the tip vortex. This, however, is countered by the high twist. To exploit this effect even more the blade area is enlarged, despite the fact that this means more thrust and torque at r/R = 85 % in relation to the baseline configuration. A higher dimensional chord parameterization might be able to utilize



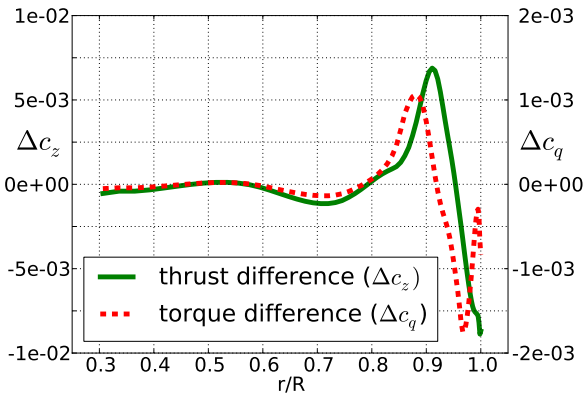
(a) Low-fidelity blade



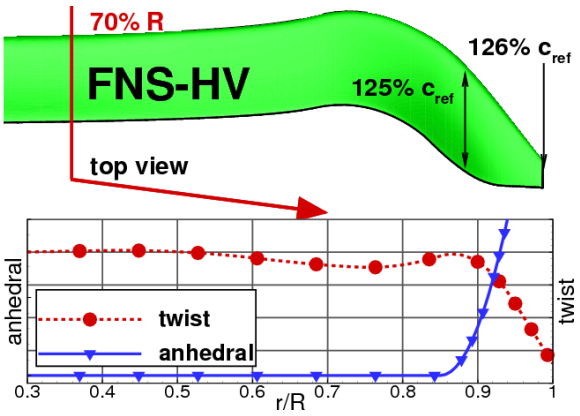
(b) Load differences (low-fidelity blade minus baseline)



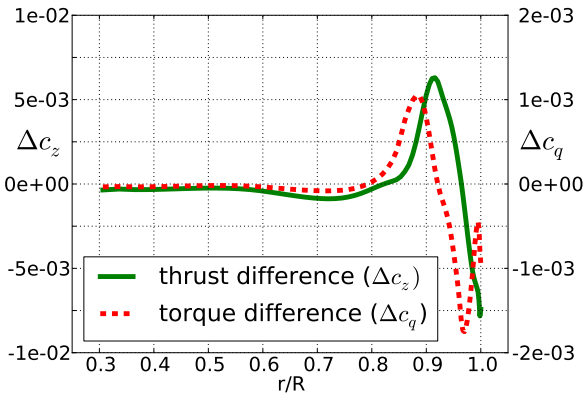
(c) Mid-fidelity blade



(d) Load differences (mid-fidelity blade minus baseline)

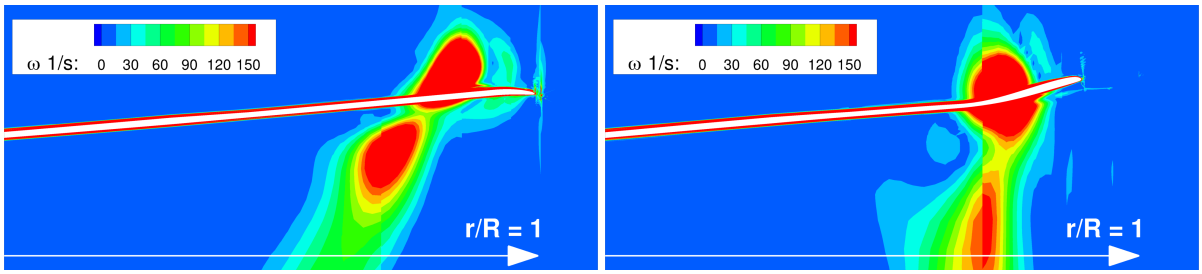


(e) High-fidelity blade in hover



(f) Load differences (high-fidelity blade minus baseline)

Figure 4.24: Best hover blades of each-fidelity for GRC optimization. Loads computed with FNS.



(a) Baseline (b) Euler optimized

Figure 4.25: Vorticity plots of base line and **EU** optimized rotor in hover. Viewed 0.2m behind the rotational axis.

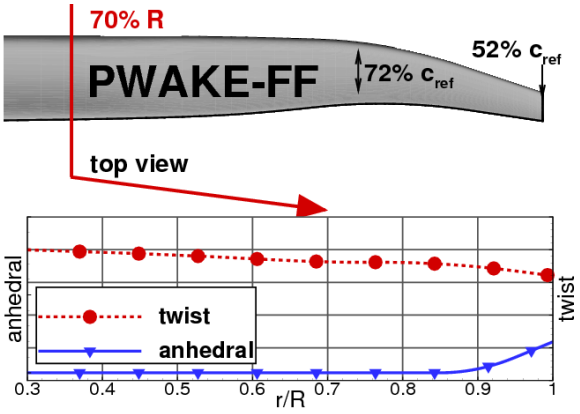
this effect even better. The anhedral present in the jig-shape mostly vanishes when the blade rotates due to the high inertial loads. Though, it does lead to an overall bigger blade flapping, almost like a pre-cone. In combination with the blade sweep, the very tip of the blade is slightly elevated. This gives the self-induced tip vortex a tendency to shift the tip vortex of the previous blade below itself increasing the exploitation effect. The vorticity is plotted in Figure 4.25(a) for the base line blade and in Figure 4.25(b) for the Euler optimized blade visualizing the phenomenon for the **EU**, but not the baseline blade.

Lastly, the high-fidelity blade is very similar to the mid-fidelity blade, with the clear distinction of a smaller tip area, which is almost equal to the baseline blade. The main reason is that due to the now resolved friction, the optimizer has to trade-off between the vortex exploit mechanism, and the general advantageous off-loading of the blade at the tip and reducing viscous drag by reducing the wetted area. For this rotor in this flight condition the optimizer choses to decrease the chord length. This effect, and a slightly modified twist, which is the most sensitive parameter for this flight condition, an additional 0.6 % reduction of required power is gained by directly optimizing with the high-fidelity instead of optimizing at mid-fidelity.

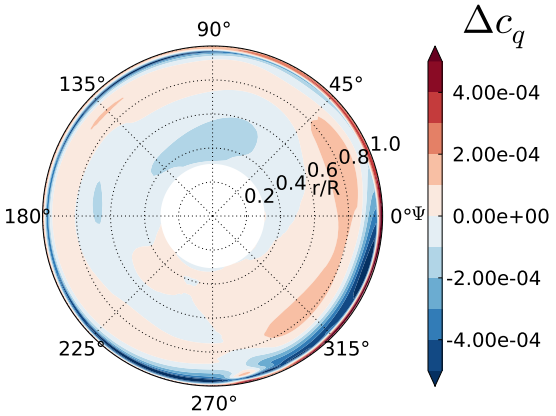
Forward Flight Optimized Blades

The blade shapes of the forward flight optimizations are drawn in Figure 4.26(a) - Figure 4.26(g) with the according torque differences in Figure 4.26(b) - Figure 4.26(h). Their predicted and with **FNS** double-checked performances are listed in Table 4.2. The conformity between the various methods is less than it is for the 7A blade. The most outstanding factor is that the mid-fidelity blades of the **EU** and **MEU** method perform worse than the low-fidelity blade. While for the 7A, both methods arrive at almost the same blade shape with similar performances, the chosen low-fidelity method **P.WAKE** actually outperforms the selected mid-fidelity **EU** when recomputed with **FNS**.

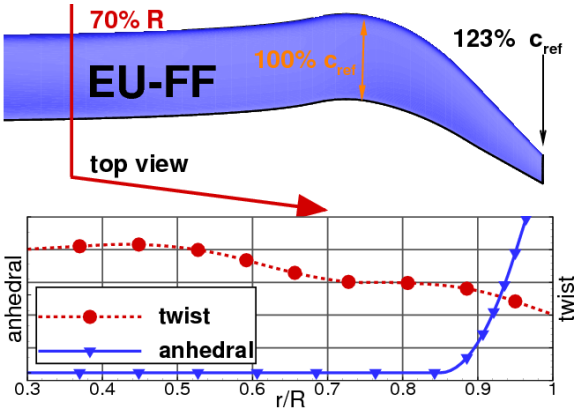
The torque distribution of the mid-fidelity **EU** blade has a ring at around r/R = 80 %, where it is heightened for every position on the blade. This ring is perceived even stronger for the **MEU** blade. This elevated torque goes hand in hand with the larger chord length at this position, which is slightly larger for the **MEU** blade. Three effects are identified for this degradation in reference to the other optimized blades. The lack of friction means no viscous drag and the lift is generated more efficiently at the 80 % radial station. The missing friction also under predicts the blade torsion as Pahlke and Wall [61] already find out. This relaxes the constraint and the **EU** and **MEU** blades



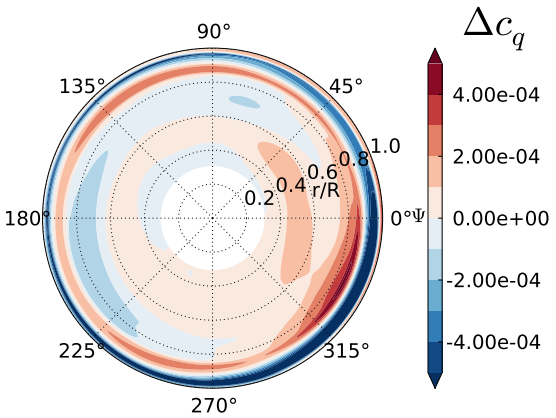
(a) Low-fidelity blade



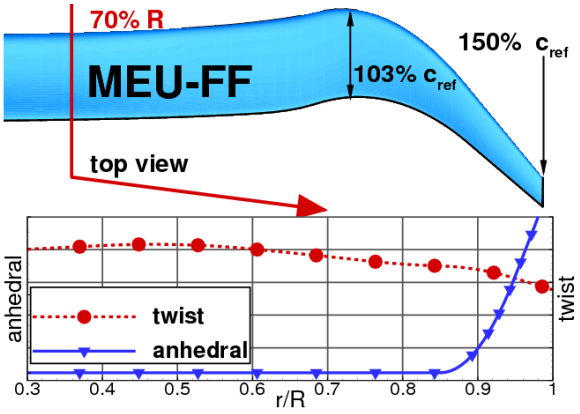
(b) Torque differences (low-fidelity blade minus baseline)



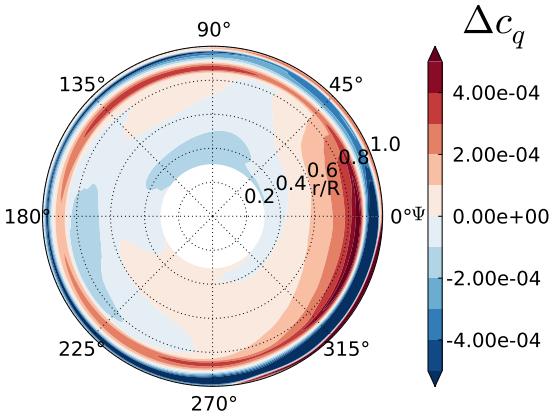
(c) Mid-fidelity blade



(d) Torque differences (mid-fidelity blade minus baseline)



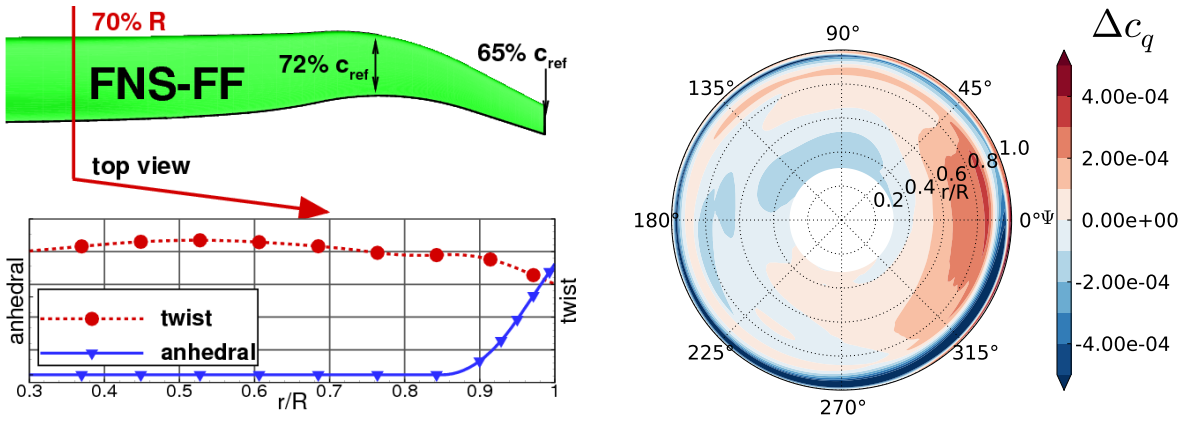
(e) High/mid-fidelity blade in hover



(f) Torque differences (high/mid-fidelity blade minus baseline)

Method	power reduction		pitch link loads reduction	
	predicted	checked	predicted	checked
P.WAKE	-5.7%	-5.0 %	-5.4%	+3.0 %
EU	-2.9%	-2.0 %	-18.8%	-14.5 %
MEU	-5.0%	-3.2 %	-11.9%	-10.0 %
FNS	-5.9%		-11.5%	

Table 4.2: Power reduction of blade of individual **forward flight** optimizations. The predicted reduction is given at the individual fidelity, the checked reduction is the result when re-computed at high-fidelity (**FNS**).



(g) High-fidelity blade in hover

(h) Torque differences (high-fidelity blade minus baseline)

Figure 4.26: Best forward flight blades of each-fidelity for GRC optimization. In the torque plot blue means reduction of torque with reference to the baseline, red an increase. Torque differences computed with **FNS**.

are swept stronger than the other two blades of **P.WAKE** and **FNS**. At the same time, the vortices are not resolved in the single blade approach. As vortices do have an impact on the inflow, they also drive the twist distribution, which is now parameterized with more parameters. Thus, the twist distribution is more receptive to local inflow changes. The latter effect becomes more important for this rotor in contrast to the 7A, as this rotor flies at a lower advance ratio and has one more blade. The lower flight velocity keeps the vortices in the system longer and the additional blade decreases the spacing between the blades. More blades weaken the individual tip vortices, but the interaction with them grows. As vorticity is captured by the **MEU** method, the twist distribution of this blade agrees better with the one from the **FNS** blade, while the **EU** twist distribution matches less. Looking at the achievable power savings with these methods, the correlation between predicted and with the **FNS** double-checked results differs noticable. The more sophisticated method **MEU**, which includes the tip vortex, allows a better power reduction than the **EU** method, however at a smaller scale than the low-fidelity method **P.WAKE**. It is concluded that it is necessary to capture both effects, viscosity and vorticity properly to grant the best blades.

The low-fidelity model **P.WAKE** features both physical phenomena, the vortices and the friction. Though the vortices are inviscid and the friction only comes in from airfoil tables, the agreement is better with the **FNS** blade than with the mid-fidelity blades of **EU** and **MEU**. At around $\psi = 280°$ and r/R = 95 % in the torque plot Figure 4.26(b)

of the **P.WAKE** blade, a little red spot is located among the otherwise blue line. After checking the flow solution of the **FNS** computation, a small flow separation is identified at the tip, which is not resolved throught the 2D airfoil tables of the **P.WAKE** method. This small separation does not have a big impact, though the pitch link moment is greater in comparison with the high-fidelity optimized blade. The minor discrepancies of high- and low-fidelity might become larger at greater thrust settings. An unexpected outcome is that the twist gradient is reversed for the **FNS** blade at the blade root. This arises from the not modeled rotor hub and the attachment points, which then predicts a different root vortex. While it does exist, it likely behaves differently and this effect is considered mostly an artifact.

4.2.3 Test-Matrix

A coarse overview of the following tests is sketched in Figure 4.27. In contrast to the 7A optimizations, the presented test matrix is thinned out. For example no mid-fidelity multi-objective tests are performed as from the single-objective mid-fidelity tests a greater loss of correlation between low- and mid-fidelity is found for the higher dimensional design space than for the four dimensional space of the 7A rotor. The high-fidelity forward flight optimization is extended with multi-blade Euler computations **MEU**, while for the multi-objective high-fidelity optimization only one trend function setup is analyzed. In hover, only the **EU**-**FNS** combination is examined for the high-fidelity optimization.

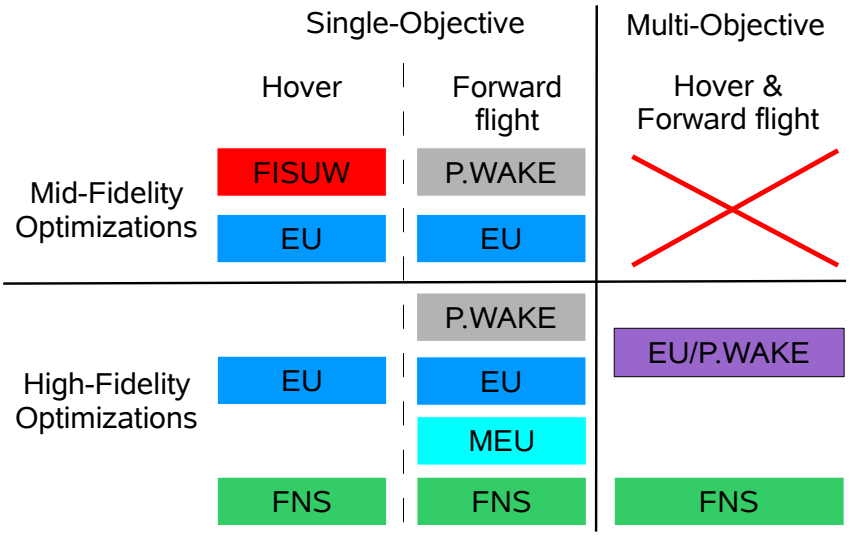


Figure 4.27: Test matrix of GRC rotor optimization

4.2.4 Single-Objective Optimizations

The single-objective optimization has a different purpose for the GRC rotor optimizations. For the 7A rotor, the potential optima are known from the genetic optimization from Section 3.3, yet this possibility does not exist anymore for the GRC rotor. The cost to perform a direct genetic optimization on a ten dimensional parameter space is beyond the scope of practicability. Therefore the single-objective optimizations reveal the best configurations of each goal function and at the same time validate the anchor

points of the following multi-objective optimizations. The more important question for the GRC rotor is whether or not the methodology derived from the 7A rotor is also valid under the new conditions and constraints added for the GRC rotor.

Mid-Fidelity

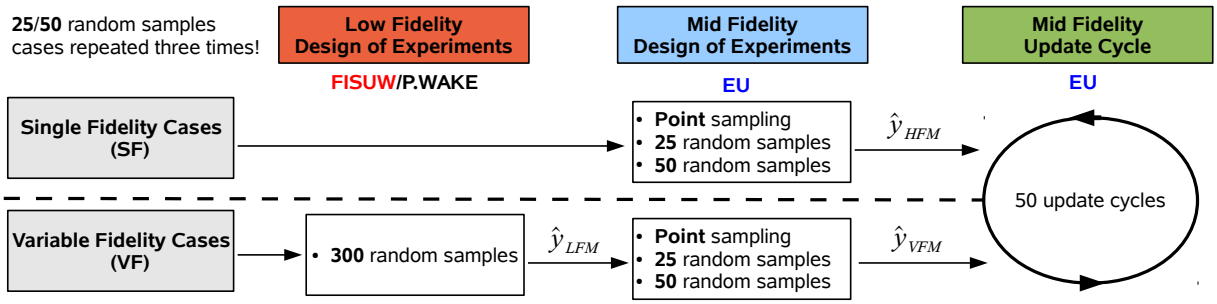


Figure 4.28: Test setup for the GRC mid-fidelity single-objective tests.

Similar to the 7A blade, the mid-fidelity optimizations help identifying the strengths and weaknesses of the (variable-fidelity) optimization framework, as well as to sight worthy optimization settings for the high-fidelity optimizations. The resources granted to each optimization case can be extracted from Figure 4.28. The variable-fidelity optimization is supported by 300 random low-fidelity samples. Both fidelity modes are then run with three different mid-fidelity design of experiments, a single point, 25 and 50 random samples. The common update cycle features 50 sequential samples. All cases are run three times in order to quantify the effect of randomness for the higher dimensional parameterization.

Optimization **Strategy 2** from Section 2.4.1 is now applied for the single-objective optimization. It employs the evolutionary algorithm instead of the very deterministic and brute force approach of a full factorial design. Latter cannot be applied as this would require to evaluate 10^8 samples, a number that is even too large for surrogate models. For the variable-fidelity optimizations containing the most number of samples in total, it is observed that about $80,000$ function calls to the surrogate models are made during the Pareto front search. This yields a run time of about 3-4 minutes on a single core computer. The surrogate settings are a search range for the hyper parameter $\theta = [10^{-4}, 10^{1}]$ and a variable λ between $[10^{-15}, 10^{-7}]$. The tuning process is also done with **Strategy 2**.

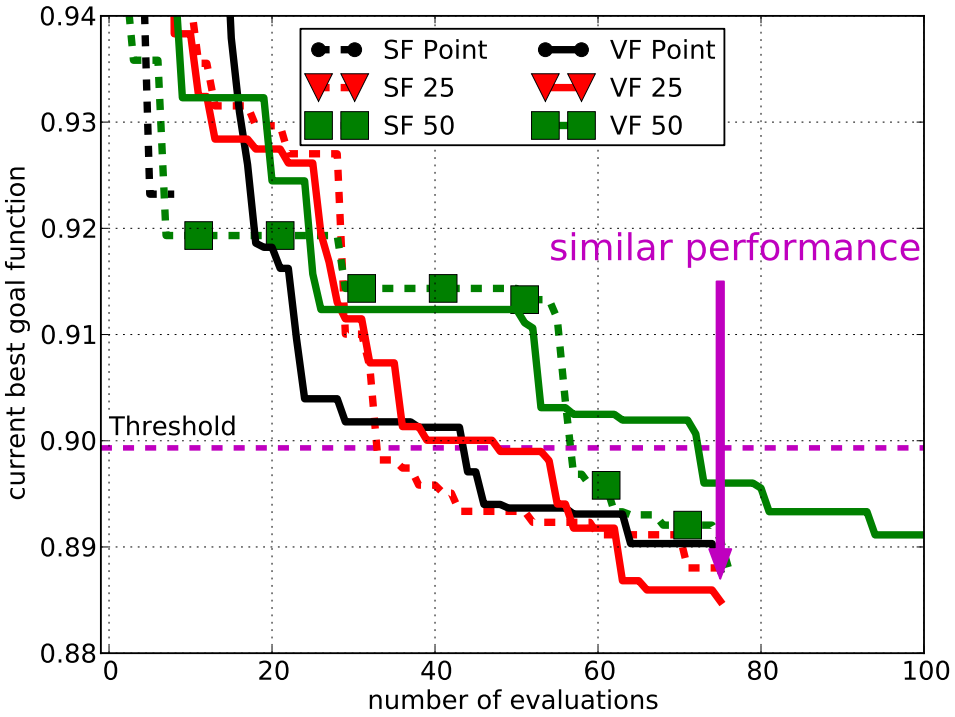
Within the optimization, the finite-state enhanced blade element theory is used as the low-fidelity method in hover (**FISUW**), while in forward flight the blade element theory with the prescribed wake model (**P.WAKE**) is applied as the low-fidelity method. The mid-fidelity Euler simulation consists of a periodic mesh containing 120,000 cells in hover and 130,000 cells in forward flight. The mesh topology is based on the 7A setups with a slight increase in grid density in front and behind the blade to keep the vortices better.

The parameter are limited and kept the same for both flight conditions. The boundaries are changed later on for the high-fidelity to match the then occurring optima. This time, the optimization is not run until all resources are spent, but stopped after it claimed to have reached a design confidence of 90% or above for three consecutive updates.

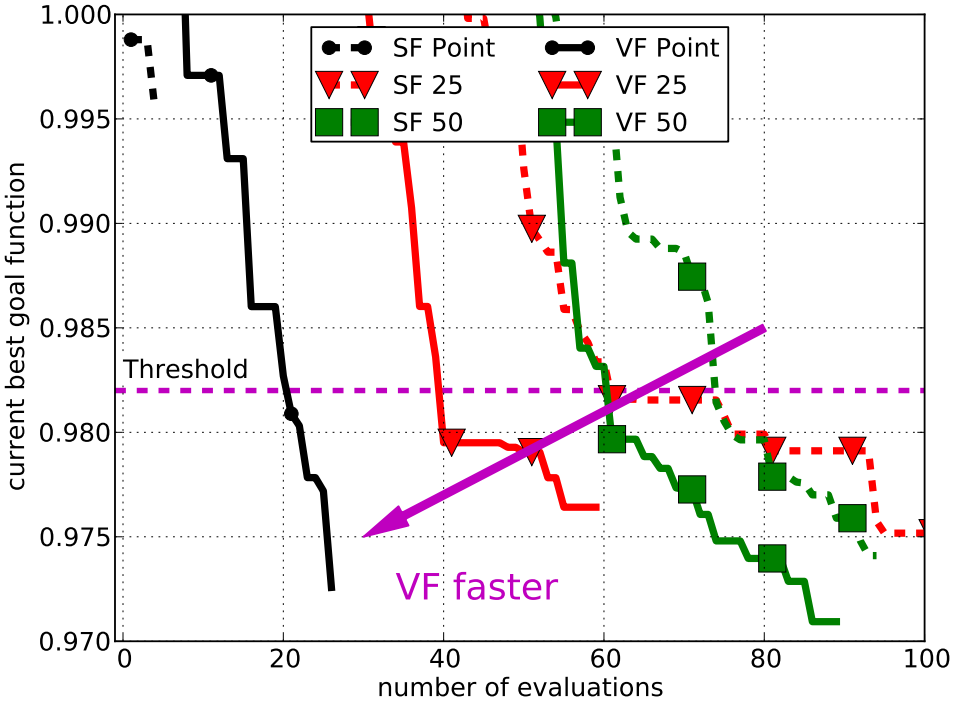
The convergence histories of the hover and forward flight optimization are plotted in Figure 4.29(a) and Figure 4.29(b). The required costs are presented in Figure 4.30(a) and Figure 4.30(b). Unlike the 7A scenario, a few optimizations fail to converge within the given resources meaning that they do not recover a configuration within an acceptable goal function value. The **SF Point** setup failed in hover as well as in forward flight. The gray bar in the charts visualizes this failure. For the **VF Point** case in forward flight, one out of three failed, which is not particularly highlighted.

Looking at the hover cases, the threshold value is set to be 0.899, 0.025 worse than the overall best configuration found of all optimizations. This is set well above the theoretical margin of 0.01 given by the design confidence criterion. Otherwise more than half of the optimizations would need to be considered invalid. However, checking the final design vectors, the found designs are similar to the configuration sketched in Figure 4.24(c). As expected, the variable-fidelity does not accelerate the optimization, which is clearly perceived from Figure 4.29(a). The **FISUW** model is too inaccurate to resolve the meticulous physics required for this type of optimization, therefore it cannot support the mid-fidelity **EU** model in the variable-fidelity process. The **SF 25** and **SF 50** cases are faster in relation to their counterparts **VF 25** and **VF 50**. When considering a different threshold at 0.89, the **VF 25** case could be better, but then the **VF Point** and **VF 50** tests need to be considered stalled. The only faster variable-fidelity setup is the **VF Point**, which is successful unlike the **SF Point** setup. Apparently the given optimization problem is saturated with 25 samples and for larger design of experiments the low-fidelity actually slows down convergence due to the larger initial costs. Nevertheless, if the most efficient single-fidelity case **SF 25** is compared to the best variable-fidelity run **VF Point**, then cost savings of 14.6% and 6.7% are found for reaching the threshold or the design confidence, respectively. The reduced correlation of low- and mid-fidelity is observed in the previous section and stresses the findings made with the synthetic test function from Section 2.6. There it is demonstrated that a false trend function costs the same or more than a regular single-fidelity optimization.

Considering the forward flight scenario; the threshold is set to 0.982, 0.015 worse than the best found shape, slightly tighter than for the hover case, yet above the design confidence margin of 0.01. A higher value would have meant that no improvement at all in contrast to the reference rotor is allowed and thus this value is chosen as a trade-off between failing too many optimization runs and still being able to make a valid statement. A first note is the high ratio of non-converging trim solutions. For the **SF Point** case it is 50.0%, while for the **SF 25** it is 15.7% and the lowest fraction has the **VF Point** case with 9.4%. The crashmap works better in the **VF Point** case due to the fact that the low-fidelity surrogate model grants the optimization a better insight as to where not to place samples. For the other optimizations, the rough resemblances given from the design of experiments suffices at this point to keep the number of erroneously evaluated samples low. The better alignment of low- and mid-fidelity in forward flight in reference to hover is reflected in the ability of the variable-fidelity process to speed up the optimization. Considering only cases where all three runs have been successful, the **SF 50** case is the fastest method to reach the threshold on the single-fidelity side, while the **VF 25** is the fastest on the variable-fidelity side, which is summarized Figure 4.30(b). These two cases are also the fastest runs to reach the design confidence in their respective fidelity mode. This leads to potential savings of 41.4% and 36.0% for the threshold and the design confidence. This is of noticeable magnitude, but in comparison to the mid-fidelity optimization for the 7A rotor, the savings are strongly reduced. This shows that for a more in-depth design, the agreement of the fidelities



(a) Hover (threshold is 0.899)



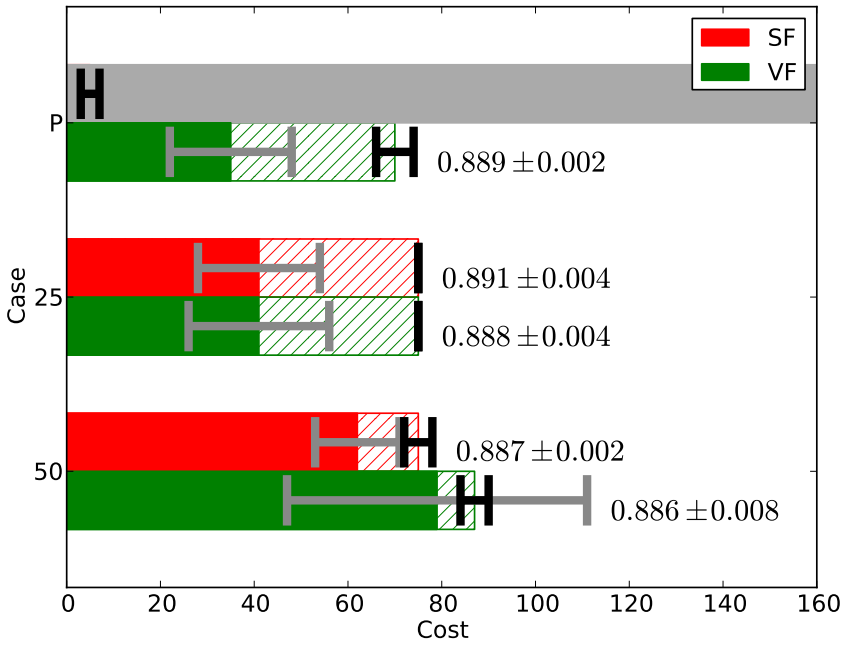
(b) Forward Flight (threshold is 0.982)

Figure 4.29: GRC single-objective optimization at **mid-fidelity**: Convergence of single- (SF) and variable (VF)-fidelity for single-objective. Goal function values averaged over three runs.

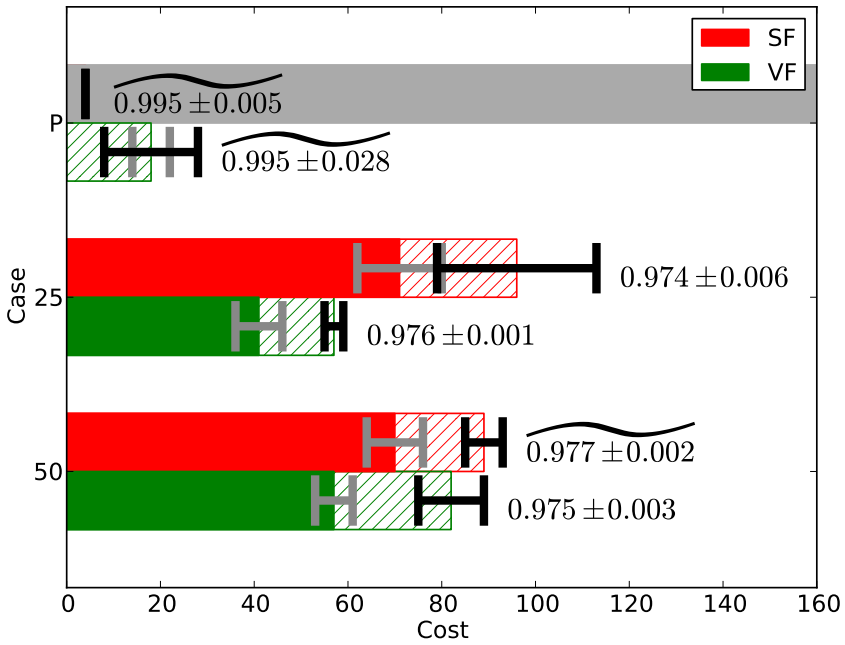
also has to become better; otherwise no gain of the variable-fidelity is obtainable.

Setting the hover and forward flight optimization into relation, the standard deviation of reaching the threshold is smaller in forward flight, but larger for the design confidence. Two reasons are identified for that. Despite the fact that the design space is restricted to a specific sweep combination, multiple minima exist in the design space for the hover goal function. This makes it more difficult to find the optimum. However, lucky shots from the design of experiments eventually shortcut the optimization. Contrary to the hover goal function, in forward flight only one global optimum exists. However, finding this optimum is equally costly in terms of iterations. The cost for the certainty varies, as the forward flight goal function is noisier, which is deduced from higher Λ values in the surrogate models. In hover, $\lambda = 1.7 \cdot 10^{-10}$ on average, while in forward flight it is $5.8 \cdot 10^{-9}$, which is then reflected by the standard deviation for the iteration of reaching the design confidence. The reason why the forward flight function is noisier is traced back to the more difficult trim procedure, which sometimes causes a rotor simulation to take up all granted coupling iterations without arriving at the specified residual.

Summarizing the mid-fidelity single-objective optimizations for the GRC rotor, it is concluded that the gain of variable-fidelity becomes smaller compared to the 7A optimization. This is attributed to the difference in the goal function landscape between low- and mid-fidelity, which grows larger with the parameter space. A saturation of samples is also observed for the hover optimizations. 50 samples in the design of experiments grant sufficient knowledge of the performance landscape and thus the underlying trend of the low-fidelity becomes unnecessary.



(a) Hover (threshold is 0.899)



(b) Forward Flight (threshold is 0.982)

Figure 4.30: GRC single-objective optimization at **mid-fidelity**: Cost to reach either threshold (filled area) or reaching DC criterion (hatched area) including standard deviation bars. The numbers are the goal function values upon reaching DC with their respective standard deviations. SF = single-fidelity, VF = variable-fidelity. Values averaged over three runs.

High-Fidelity

The high-fidelity investigations of the ten dimensional design space for the GRC rotor are not examined with as many test cases as it is done previously with the mid-fidelity optimizations. The sole reason for this is the high cost associated with it. Just as with the 7A high-fidelity optimizations, the focus is laid more upon investigating fidelity combinations rather than the effect of differently sized design of experiments.

Keeping in mind the following multi-objective optimizations, the data generation of the low-fidelity trend functions as well as the high-fidelity design of experiments is performed in a multi-objective fashion first and later on split up for the single-objective optimizations. The process is explained in further detail in Section A.3.

Starting with the hover setups, which are on top of Figure 4.31; the single-fidelity is directly starting with 50 random samples from the design of experiments. For comparison, two variable-fidelity cases are launched. The first one, **VF Two Points + 30**, is constructed by the multi-objective data. It contains 300 random samples from the **EU** method, with an additional 100 samples from ten update cycles that execute ten samples in parallel. For the high-fidelity design of experiments, the two anchor points are chosen along with 30 random samples. The **VF Point + 10 DSS** case is specifically tailored towards the single-objective hover optimization. A mid-fidelity hover optimization is executed first, which has a size of 50 random samples in the design of experiments and another 50 samples for the update cycle. Based on this mid-fidelity optimization, the high-fidelity design of experiments samples the mid-fidelity optimum along with ten random samples (every third sample of the 30 random samples from the **VF Two Points + 30** case). This case distinguishes itself from the other cases, as it then shrinks the design space around the mid-fidelity optimum, which is also the so far best high-fidelity sample, according to the mechanism described in Section 2.5.3. In all fidelity modes, a total of 100 update cycles is granted.

The convergence of the hover optimizations is plotted in Figure 4.32(a), while the total cost of these optimizations is displayed in Figure 4.33(a). It is seen that the variable-fidelity setup with the locally pre-refined and shrunk design space case **VF Point + 10 DSS** is outperforming the other two setups. The reason is that the obtained mid-fidelity rotor configuration is obtained by a pure mid-fidelity hover optimization and not a mid-fidelity multi-objective optimization as is the case for the **VF Two Points + 30**. The deviation of this mid-fidelity rotor configuration from the high-fidelity optimum is less than 1.0% in rotor performance. Thus, the optimization starts below its theoretical margin of 0.01 for the design confidence, which is also considered as the margin for the threshold. Not only does this reduce the overall cost, but also the fact that the trend generation is roughly three quarter cheaper than for the **VF Two Point + 30** case. In opposition to the good convergence, the single-fidelity barely reaches this optimum within the allocated resources and finishes just within the allowed margin, though it itself does not claim a sufficient design confidence. In terms of total cost of the optimizations, the slim variable-fidelity setup **VF Point + 10 DSS** is significantly decreased in cost. While the threshold value, when generously specified, allows the single-fidelity to achieve this, the variable-fidelity is already done after the design of experiments. This results in a reduction of resources of 85.3% for the **VF Point + 10 DSS** case in contrast to the single-fidelity and about half of the resources compared to the larger **VF Point + 30** case. An acceleration of over 54.0% for the **VF Point + 10 DSS** case, and 31.0% for the **VF Two Points + 30** case is gained for fulfilling the design

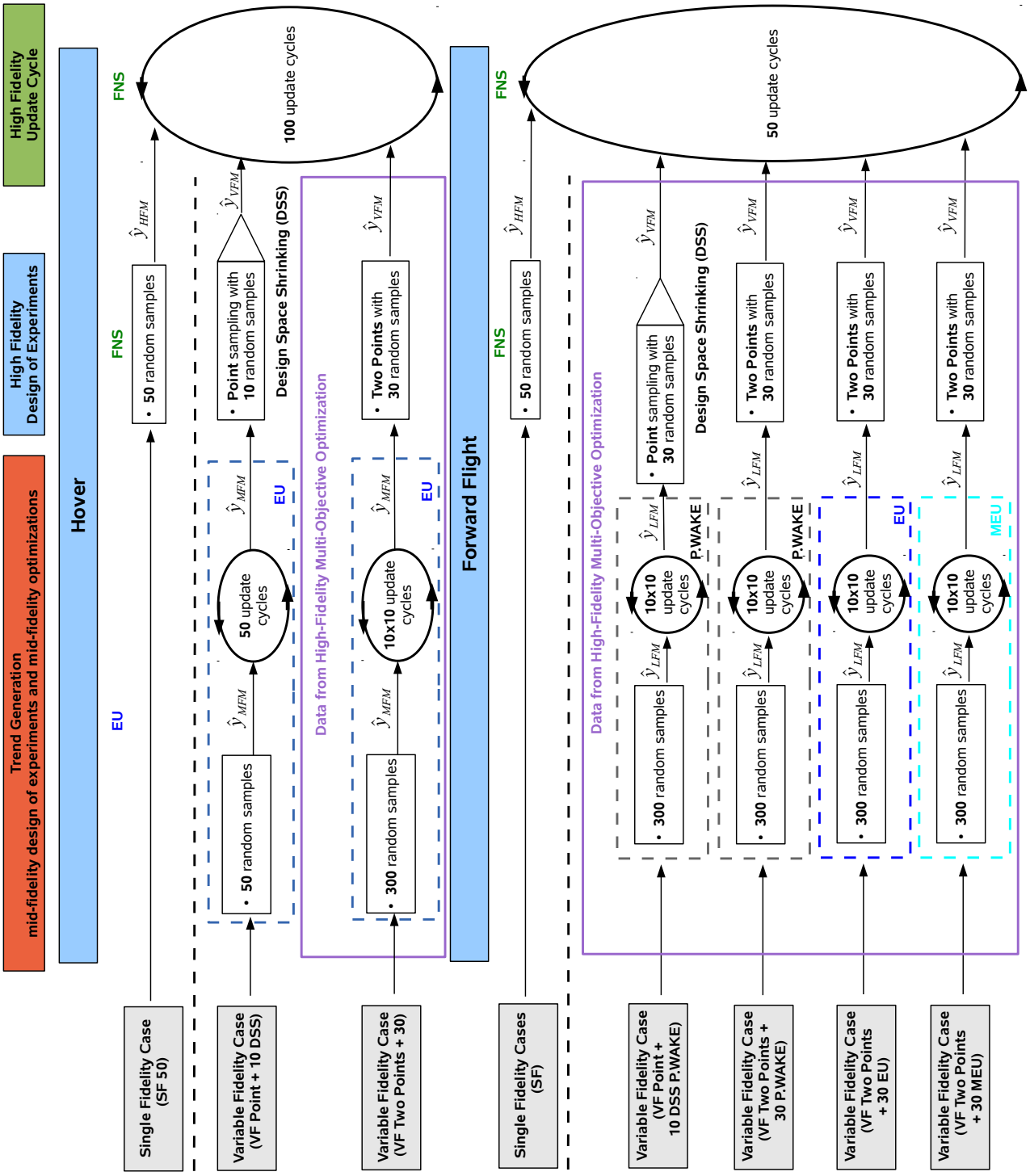
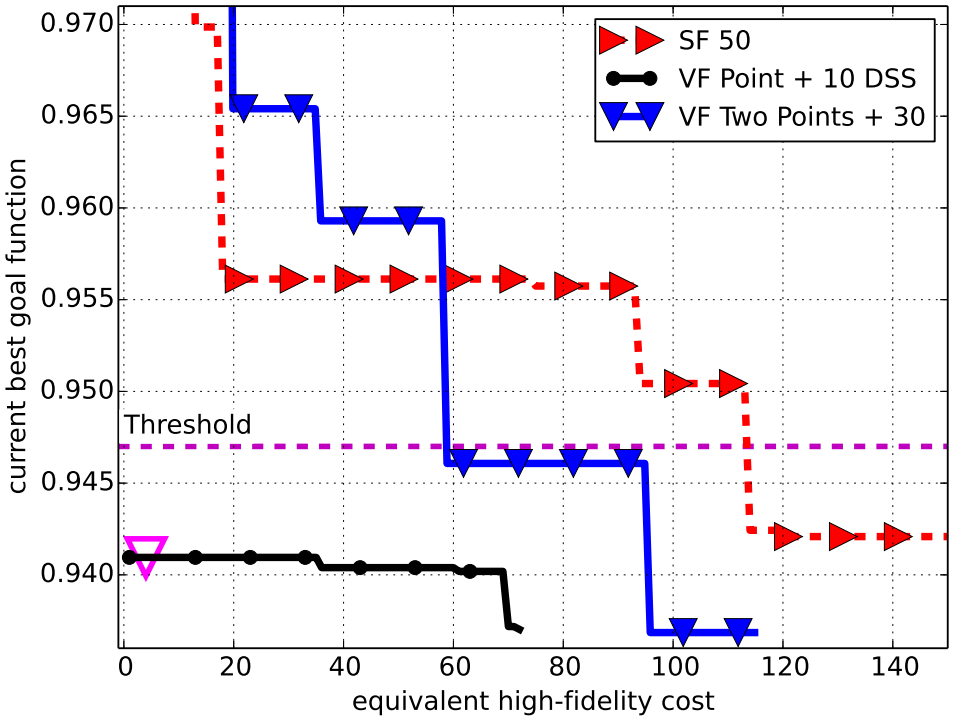
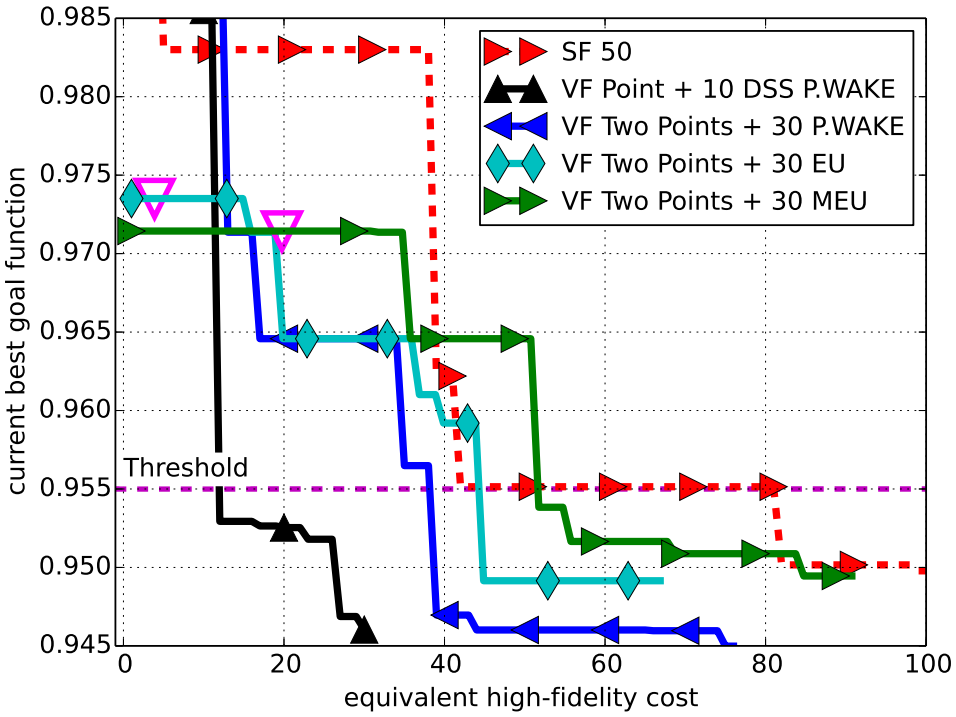


Figure 4.31: Test setup for the GRC high-fidelity single-objective tests.



(a) Hover (threshold is 0.947)



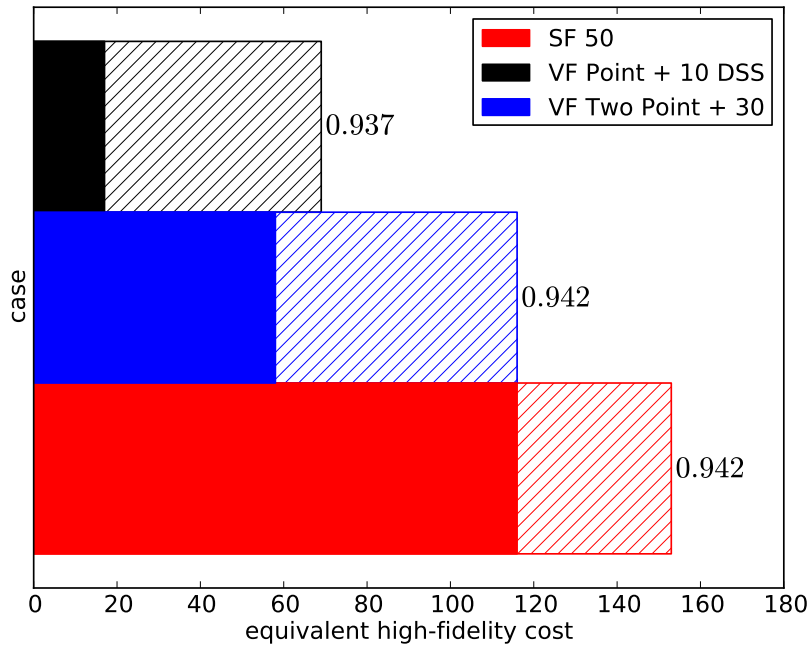
(b) Forward Flight (threshold is 0.955)

Figure 4.32: GRC single-objective optimization at **high-fidelity**: Convergence of single- (SF) and variable (VF)-fidelity. The ▽ represents the initial low-fidelity costs.

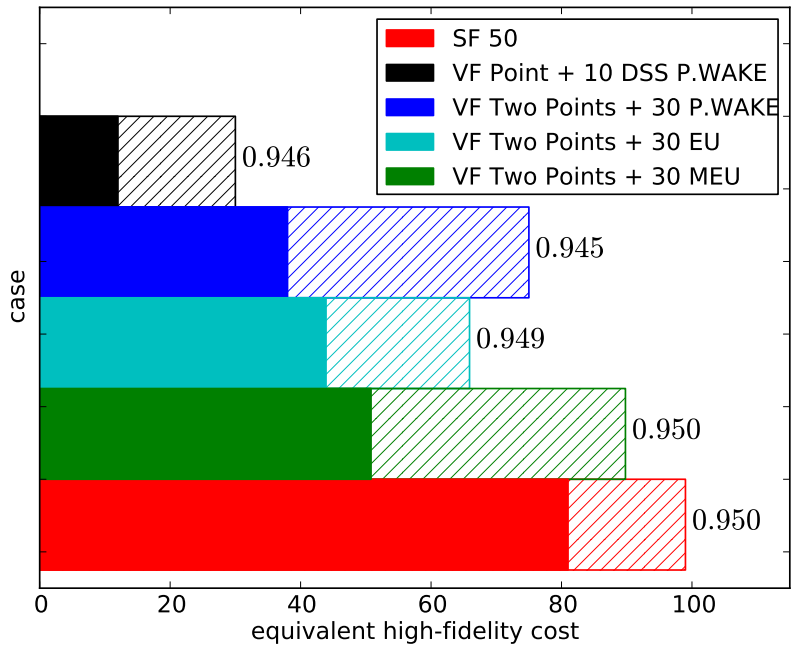
confidence criterion when compared to the total amount of resources granted to the single-fidelity approach. The necessary number of updates becomes larger as opposed to the mid-fidelity hover optimizations. This is an indicator that the goal function is more complex at high-fidelity level. Indeed, the added viscosity weighs the chord parameter differently. It flattens the influence of both chord related parameters in the performance landscape, but raises it for the constraint on the pitching moment. Both effects make it more difficult to quickly find the optimum in contrast to the mid-fidelity optimization.

The forward flight scenario features more variable-fidelity cases. Reason for this is that the best matching trend function is to be identified. Three of the variable-fidelity cases are set up similarly to the **VF Two Points + 30** case from the hover optimization with the exception that besides the **EU** method, the **P.WAKE** and **MEU** method are tried out. The **MEU** method is a multi-blade Euler simulation of the GRC rotor. It utilizes the same blade meshes as the **EU** method and additionally convects the wakes and vortices of the blades in a background mesh of $740,000$ cells. As the degrees of freedom grow with these three trends, the cost greatly varies depending on the method. While the prescribed wake model **P.WAKE** does not cost anything in comparison with a high-fidelity simulation, 0.002 high-fidelity samples, the single blade trend **EU** costs 2.87 high-fidelity samples, and the multi-blade **MEU** surrogate generation requires resources equivalent to 18 (!) high-fidelity samples. On top of these three variable-fidelity cases, a more cost efficient case is analyzed. For the **VF Point + 10 DSS P.WAKE** case, the high-fidelity data from the design of experiments of the **VF Two Points + 30 P.WAKE** case is stripped down to eleven high-fidelity samples. The first sample is the low-fidelity optimum and ten samples are from the random pool. To further streamline the optimization, the design space is shrunk after the high-fidelity design of experiments.

The optimization convergence of the GRC forward flight optimizations (Figure 4.32(b)) is faster with the variable-fidelity approach in contrast to single-fidelity approach, close to the behavior of the 7A high-fidelity forward flight optimizations. The trend based upon the prescribed wake model **P.WAKE** dominates the optimizations overall and finds a rotor configuration that none of the other versions based on **EU** and **MEU** retrieve. Except for the single-fidelity, all cases reach their design confidence. The threshold is set to 0.01 worse than the best point, which is of the magnitude of the margin for the design confidence. The single-fidelity almost passes this threshold with the 41st sample, which is part of the design of experiments and thus considered pure chance resulting from the random process. The **P.WAKE** setup with the least resources is superior for reaching the threshold and requires an equivalent of 23 high-fidelity computations. Here, the design space shrinkage is held responsible for the tremendous speed up as the optimizer is forced to search only in vicinity of the low-fidelity optimum. This is visualized in the performance landscape for the design space shrinking case for the twist_3/twist_4 slice in Figure 4.34. The gray area corresponds to the design space before the shrinkage, while the colored area is the shrunk area in which the high-fidelity optimization is performed. The red area marks the constraint violation of the pitching moment. Moving onto the costs of the forward flight optimizations charted in Figure 4.33(b), the design space shrinking mechanism grants a speed up of 85.2% for the **VF Point + 10 DSS P.WAKE** case over the single-fidelity, followed by the more costly variant **VF Two Points + 30 P.WAKE** with 53.1% reduction. The most costly variable-fidelity case, the **MEU** approach, grants a reduction of 37.3%. As for the design confidence, the **VF Point + 10 DSS P.WAKE** case is again the most efficient



(a) Hover (threshold is 0.947)



(b) Forward Flight (threshold is 0.955)

Figure 4.33: GRC single-objective optimization at **high-fidelity**: Cost to reach either threshold (filled area) or reaching DC criterion (hatched area). The numbers are the goal function values upon reaching DC. SF = single fidelity, VF = variable fidelity

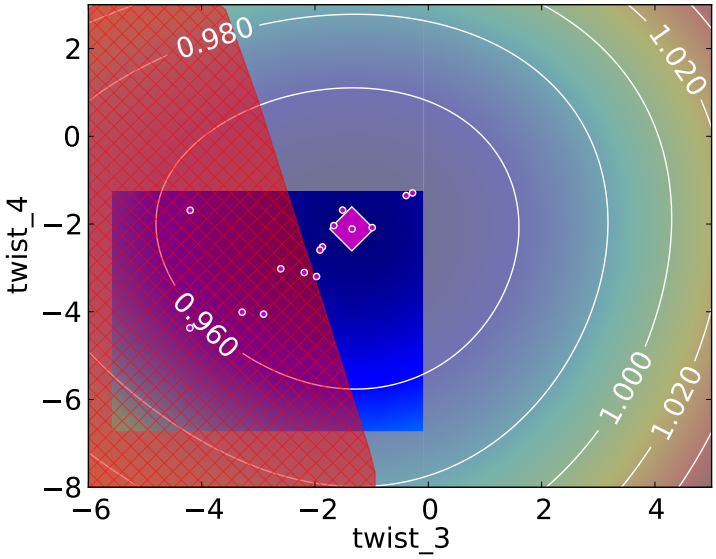


Figure 4.34: Example of shrunk design space in forward flight. The grayed area is excluded from the optimization, the red hatched area is the constraint area (XX) and the magenta diamond the optimum location ◇. The magenta dots are high-fidelity samples •.

setup with cost savings of 69.7% followed by the **VF Two Points + 30 EU** case with 33.4%. The third place goes to **VF Two Points + 30 P.WAKE** with 24.2% and last is the **VF Two Points + 30 MEU** case with 9.3%. Again, the relatively low gain from the **MEU** scenario is attributed to the high initial cost for the trend function.

This underlines that in particular for forward flight close attention has to be paid to the selection of the trend model. Not only determines the trend the number of update samples required to calibrate it to the high-fidelity behavior, but also contributes significantly to the overall cost of the optimization process. While multi-blade Euler grant good blades by itself when double checked with the Navier-Stokes solution, the cost of the simulation itself grows drastically in comparison with the single-blade approach. An additional discovery is made for the design confidence prediction, which is more accurate for the high-fidelity as it has been for the mid-fidelity optimizations. The individual runs do not stop prematurely nor is the deviation from the actual best configuration found larger than the margin specified by the design confidence criterion, which is true for all runs.

While no statistically strict quantities, especially for the design confidence criterion, can be given due to the small number of test cases performed, the advantage of the variable-fidelity approach is evident and cost savings of about a half or more of what would be paid with single-fidelity are achievable.

4.2.5 High-Fidelity Multi-Objective Optimizations

As the high-fidelity multi-objective optimization is the most cost intensive and complex task, only two cases are investigated. The flow chart of the single- and variable-fidelity optimization are given in Figure 4.35. The single-fidelity optimization **SF 50** is directly started with a high-fidelity design of experiments containing 50 random samples. In opposition to the single-fidelity optimization, the variable-fidelity optimization process **VF Two Points + 30** first samples 300 random low-fidelity samples which

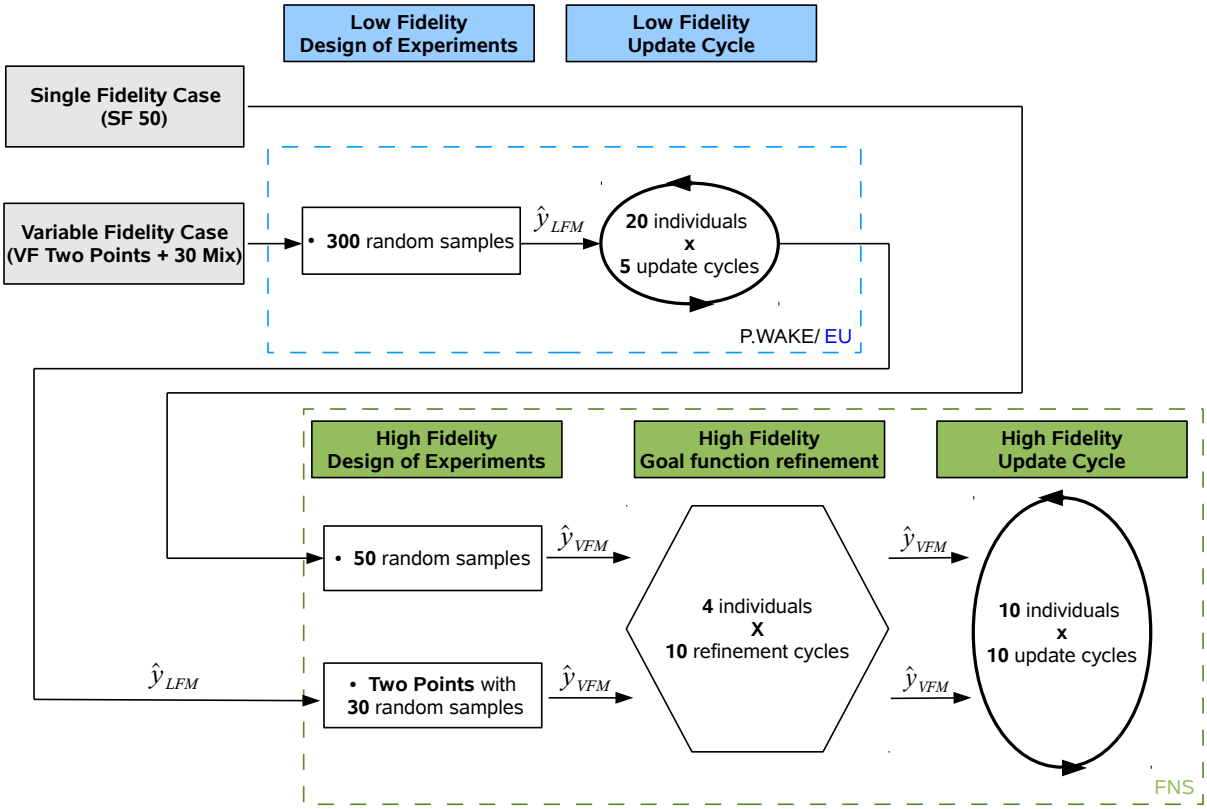


Figure 4.35: Test setup for the GRC high-fidelity multi-objective tests.

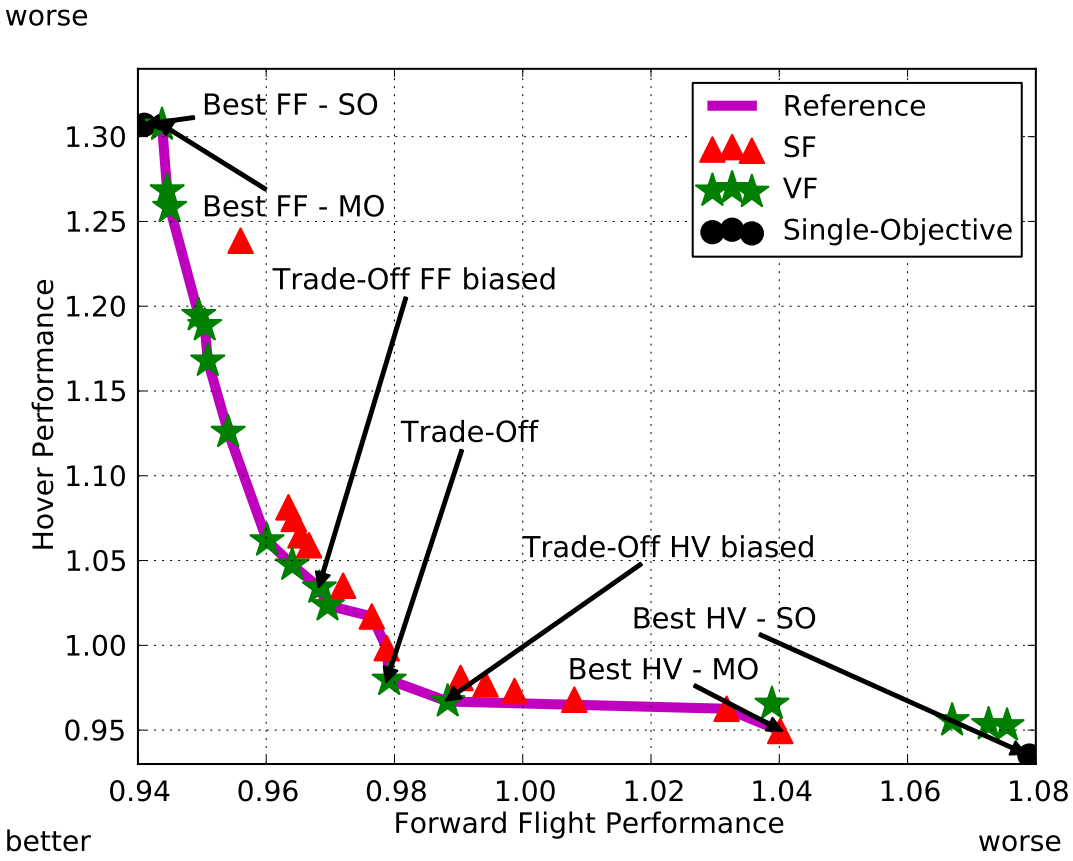
are then further refined in ten multi-objective update cycles with 10 samples each. The chosen low-fidelity is a mix of the low-fidelity model **P.WAKE** in forward flight and the mid-fidelity model **EU** in hover. The high-fidelity sample design of experiments then consists of the two anchor points found in the low-fidelity surrogate model along with 30 random samples. This roughly yields the same price for the initial trend generation for single- and variable-fidelity cases. Both cases then proceed with the goal function refinement. A total of four samples is selected for ten cycles. The four samples are chosen based on the best found goal function value and expected improvement for each goal function, while respecting both pitching moment constraints, the constraint from hover as well as forward flight. This ensures that only rotors valid in both flight conditions are inserted into the surrogate models, while others are added to the crashmap, see Section 2.5.4. After both goal and constraint surrogates are refined in the areas of the individual optima, the high-fidelity multi-objective update cycle is started. It runs for ten cycles with selecting a maximum of ten samples. The results of these optimizations are also published in [4] by the author.

Looking at Figure 4.36(a) the Pareto optimal set (—) obtained from the high-fidelity samples belonging to both cases are connected by the magenta line and is used as the reference solution. It does not represent the true Pareto front, as it would consist of an infinite number of points in this case. The points of the Pareto front reported from the variable-fidelity (∗) case are marked by green stars, while the single-fidelity (△) ones are red triangles. The best single-objective rotors (•) from the hover and forward flight optimizations are marked by black bullets (Best HV - SO and Best FF - SO). Additionally, the trade-off rotors discussed later in Section 4.2.6 are highlighted by the arrows. The single-fidelity features the best found hover rotor from the multi-objective simulation and two more Pareto optimal samples towards the hover goal function. Unfortunately neither methodology found the best hover rotor from the single-objective

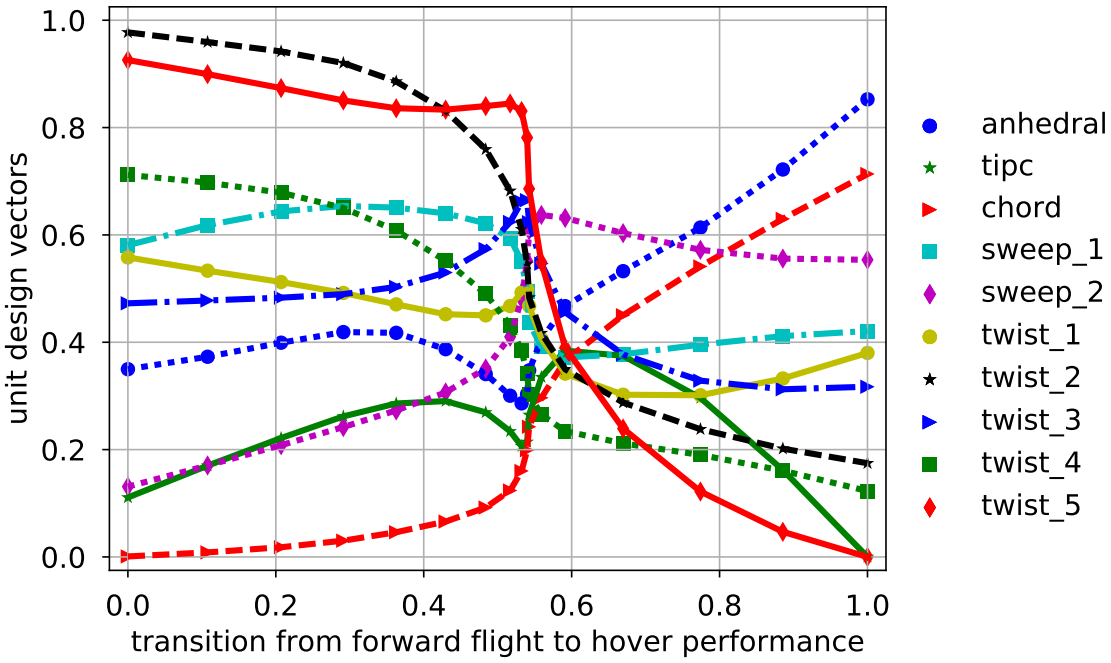
optimization, even though this rotor does not violate the forward flight constraint and should be the actual anchor point. The variable-fidelity process finds three rotor configurations of almost equal forward flight performance to the best single-objective hover rotor, but they do not reach the same performance in hover. They, however, are in the parameter area of the best hover rotor from the single-objective optimization. Reason for this is the difficulty in resolving this complex area properly along with the two constraint functions as all four surrogate models need to be sufficiently accurate here. The forward flight anchor point is better resolved, as the variable-fidelity process places a sample very close to the actual single-objective optimum. The variable-fidelity approach clearly distinguishes itself from the single-fidelity; the latter does not place a point in the vicinity of this anchor point. Comparing the number of samples contributed to the reference front by the single- to the variable-fidelity ones, then four samples are found by the single-fidelity and 13 by the variable-fidelity with a total of 17. This yields 325% more viable samples from the variable-fidelity optimization, yet this is only one (subjective) metric for a multi-objective optimization.

The complexity of the optimization problem is further understood with Figure 4.36(b). There, the parameters of the combined Pareto front are sketched, after they have been smoothed slightly for better visualization. Heilers [137] finds out that the goal function landscapes for helicopter rotors become shallow for many parameters in higher dimensions. Finding the exact path of the Pareto front becomes more difficult to resolve within the surrogate models. Yet, insight about the behavior of the parameters is extracted. For example, the tip parameter twist_5 starts out at a moderate angle for the forward flight configurations and then negatively increases towards hover flight. This behavior agrees with the previous findings from the single-objective optimizations, but the distribution between the goal functions is not known from single-objective optimizations. There also exists a kink at just past half-way towards the hover goal function, where most variables exhibit a discontinuous behavior. This is an indicator for a local minimum in at least one goal function. The Pareto front starts along from the forward flight optimum moving towards the local hover minimum to then change to the global hover optimum at half way. Such a behavior has been also observed for the 7A rotor in Section 4.1.4.

The obtained Pareto optimal set of simulated samples from each optimization is compared against the Pareto optimal set found in the respective surrogate models in Figure 4.37. The better method should have a better alignment of surrogated to simulated front. The single-fidelity plot is given in Figure 4.37(a) and the variable-fidelity plot in Figure 4.37(b). For both cases the surrogated fronts (-), green line, show better goal function values than the true samples (X), in red Xs. Two potential reasons exist for this; firstly, the simulated samples still have room for improvement, secondly, the accuracy of the surrogate models is still limited. Unfortunately the latter is the more dominant factor, because the goal function values match closely with 0.13% relative difference when comparing the best forward flight configuration from the multi-objective optimization with the one obtained from the single-objective optimization. The hover goal function has some potential left, the goal function value of this anchor point is 0.013 or 1.34% from the single-objective optimization value away. It is concluded that it is very unlikely that an improvement of more than this difference is to be expected. The dashed magenta line marks the theoretical margin of 0.01 (--) that the Pareto confidence criterion allows if a Pareto confidence of a 100% is reached, which neither case has. The line is simply generated by adding this 0.01 margin in both directions to the surrogate Pareto front.

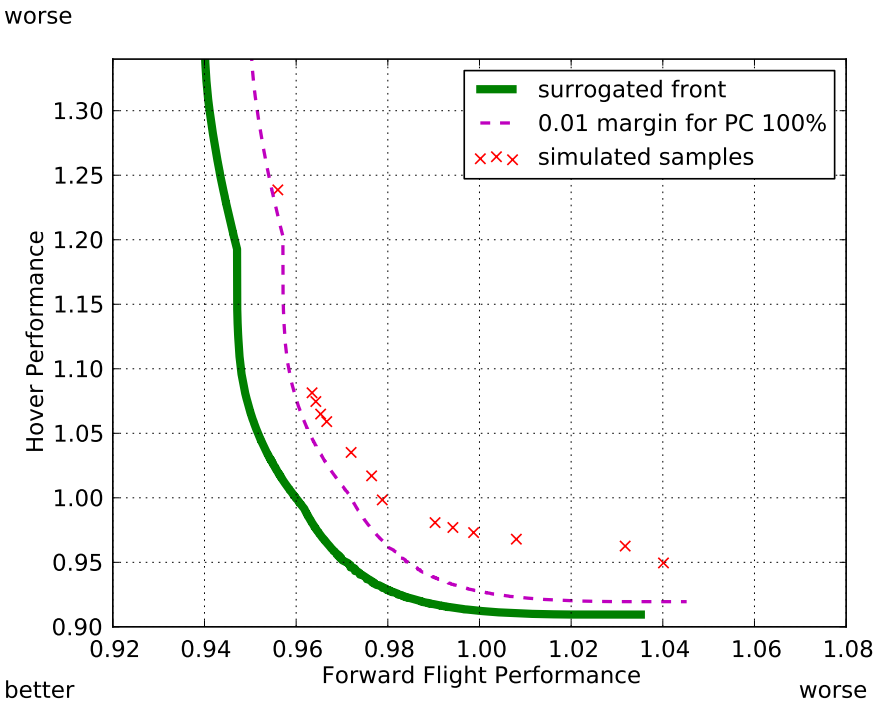


(a) Pareto front

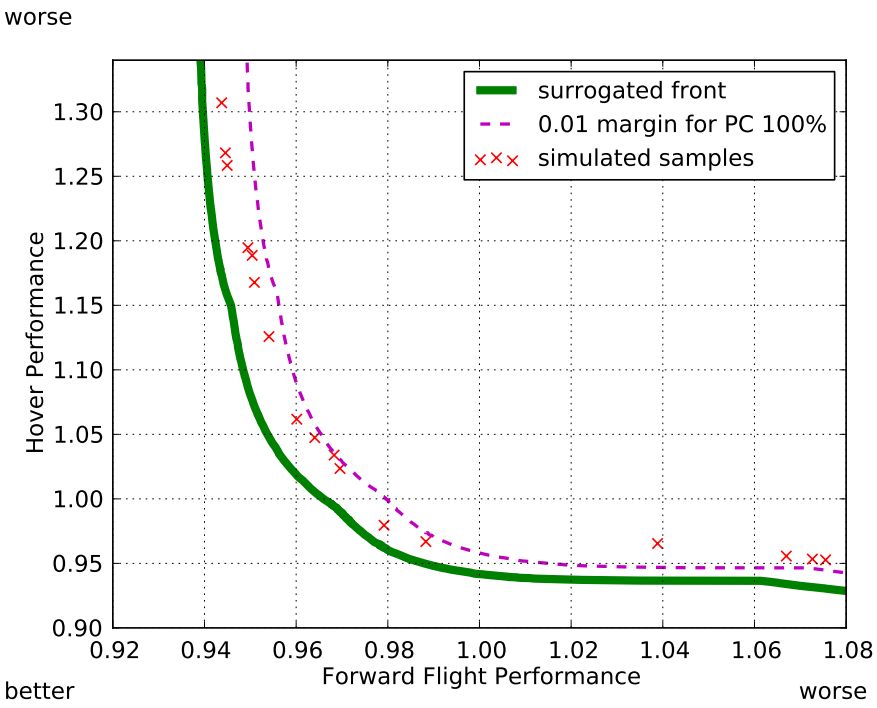


(b) Planform parameters (smoothed)

Figure 4.36: GRC high-fidelity multi-objective optimizations: Comparison of single- (SF) and variable-(VF) fidelity Pareto fronts and parameters obtained.



(a) Single-Fidelity



(b) Variable-Fidelity

Figure 4.37: GRC high-fidelity multi-objective optimizations: Comparison of Pareto optimal samples with the surrogated Pareto front approximation.

For the variable-fidelity surrogate Pareto front, all samples with a forward flight performance better than 1.0 are contained within this margin, while the other samples, which are better at hover, are not. For the single-fidelity, none of the samples is within the margin. Yet, the tendency that samples with better forward flight performance deviate less from the surrogate Pareto front is also observed. This is additionally reflected by a slightly higher Pareto confidence of 78.4% for the forward flight in contrast to and 67.4% in hover for the single-fidelity optimization. The variable-fidelity optimization features 73.8% and 60.4% Pareto confidence for forward flight and hover respectively. It is noted that the single-fidelity claims more confidence in both goal functions, but has a greater discrepancy of surrogate front and real samples. This is also an indicator for less accurate surrogate models, as the variable-fidelity claims less Pareto confidence, but has a better agreement between surrogated Pareto front and sampled Pareto front.

In order to quantify the quality of the solutions, a discrete error integral is defined. As the resulting samples do not lie on a particular grid, the functional relations of

$$\hat{y}_{hover} = f(y_{forward\ flight}) \text{ and } \hat{y}_{forward\ flight} = f(y_{hover}) \tag{4.1}$$

are generated. The surrogate model $\hat{y}_{hover}$ is then evaluated for 10,000 evenly spaced points between $min(y_{forward\ flight})$ and $max(y_{forward\ flight})$. $\hat{y}_{forward\ flight}$ is evaluated accordingly. This is done for the surrogated and simulated front. The error between surrogate and simulated front is then specified as the average of their distances:

$$\epsilon = \frac{1}{n}\sum_{i}^{n}(\sqrt{(\hat{y}_{simulated,i} - \hat{y}_{surrogated,i})^2_{hover} + (\hat{y}_{simulated,i} - \hat{y}_{surrogated,i})^2_{forward\ flight}}) \tag{4.2}$$

with n being the 10,000 samples. The error then becomes 0.136 for the single-fidelity and 0.043 for the variable-fidelity approach, which is a ratio of 3.2 in favor of the variable-fidelity surrogate models.

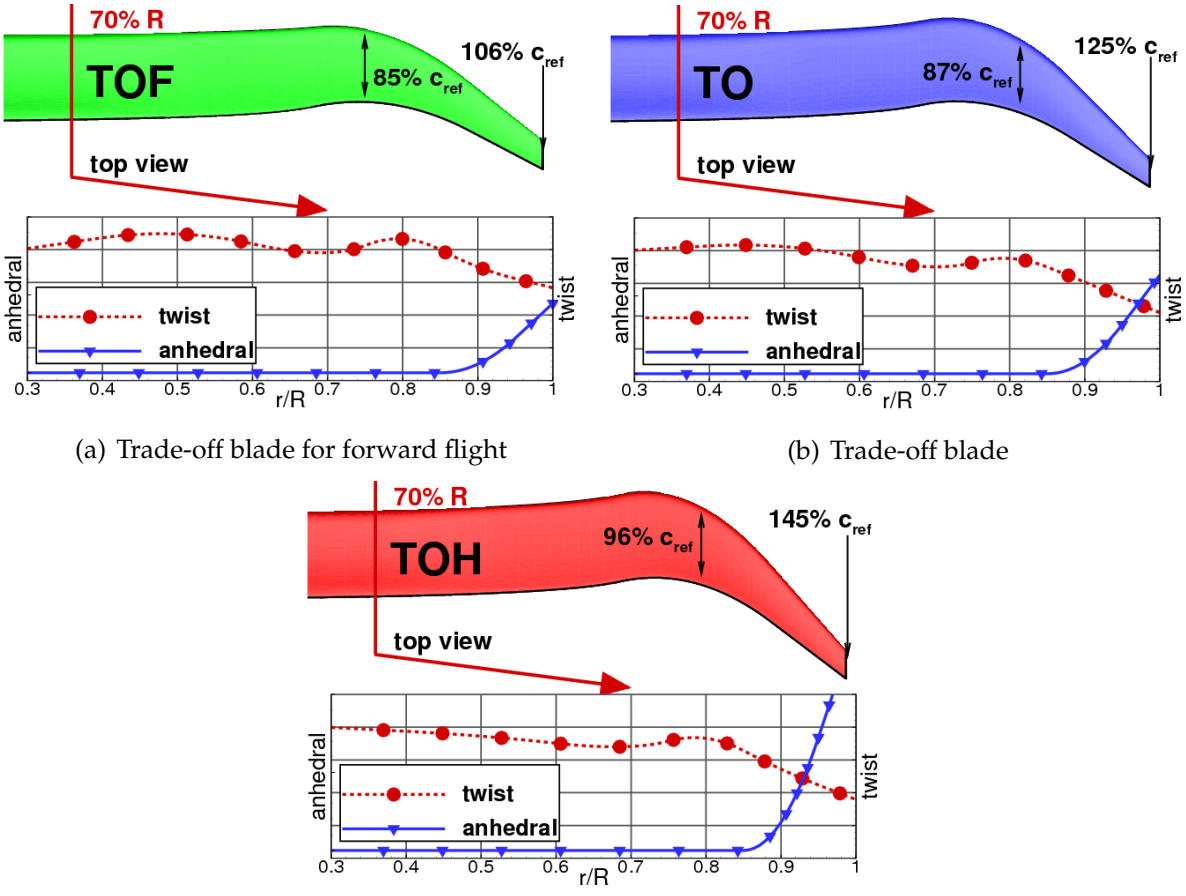
For this single test of the multi-objective high-fidelity optimization with a complex parameterization, the variable-fidelity approach proved to be of advantage. It places more simulations close to the real front, while at the same time the accuracy measured as the distance of the surrogate Pareto front to the real front is higher. Especially the latter is important, as the correct solution of the Pareto front within the surrogate model allows the optimizing engineer to find intermediate configurations without the need to run any additional simulations.

4.2.6 Analysis of Selected Rotor Designs in Off-Design Conditions

One question that arises, especially for the complex optimization, is "how much does this optimization contribute in finding a reasonable design?" On the one hand the rotors are evaluated at very specific flight conditions. This means that they may suffer strong disadvantages when deviating from this flight condition. On the other hand the multi-objective optimization finds many intermediate designs, which balance both flight conditions. Therefore, three blades are hand-selected from the multi-objective optimization at the location of the kink of the Pareto front and are marked in Figure 4.36(a). All of them represent trade-off blades. The first one is slightly forward flight biased, the second one tries to balance both goal functions and the third one is more hover biased. The off-design conditions chosen for these blades are here considered through a variation of thrust in hover and a variation of cruise velocity in forward flight. All rotors are tested in both flight conditions.

Method	forward flight		hover	
	req. power	constraint pitch link loads	req. power	constraint pitch link loads
Best FF blade (FF)	-5.9%	-12.4%	+30.7%	-23.8 %
Trade-Off FF (TOF)	-3.5%	-13.6%	+3.4%	-0.4%
Trade-Off (TO)	-2.4%	-30.5%	-2.0%	-4.2%
Trade-Off HV (TOH)	-1.5%	-30.3%	-3.3%	-0.4%
Best HV blade (HV)	+7.9%	-12.9%	-6.5%	-0.5%

Table 4.3: Improvements of selected GRC multi-objective optimized rotors. The constraint is defined as the relative difference in maximum pitch link loads of optimized rotor and baseline blade. FF = forward flight, HV = hover



(a) Trade-off blade for forward flight

(b) Trade-off blade

(c) Trade-off blade for hover

Figure 4.38: Selected blades from the GRC rotor multi-objective optimization

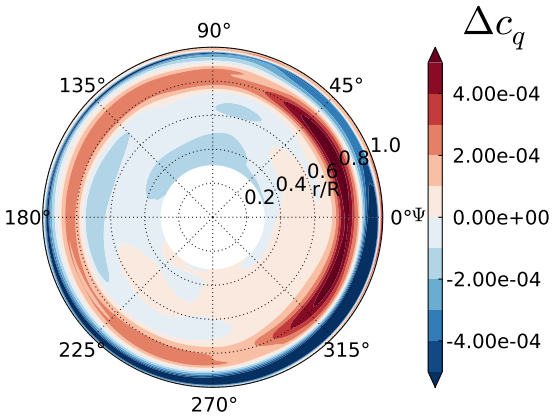
The three blades are depicted in Figure 4.38(a) to Figure 4.38(c) with their performances listed in Table 4.3. Additionally, the load differences in forward flight and hover in comparison with the reference rotor are given in Figure 4.39(a) to Figure 4.39(f). Their respective Figure of Merit polars in hover are plotted together with the reference rotor as well as the best forward and hover blade in Figure 4.40. The relative differences in the power requirements over the forward flight speed envelope are drawn in Figure 4.41.

The **trade-off** blade with **forward** flight bias (TOF) in Figure 4.38(a) has a common basis with the best forward flight blade pictured in Figure 4.26(g). A larger twist and chord length at around r/R = 80 % raise the torque over the whole disc in forward flight, Figure 4.39(a). In combination with a slight growth of the outboard sweep and twist, this channels into a better performance in hover. With the greater twist at r/R = 80 %, the twist gradient is enlarged towards the tip, which matches better with the tip vortex of the previous blade. The thrust is then generated more inboard and the outer tip is relieved, Figure 4.39(b). Looking at the hover thrust polar in Figure 4.40, the performance is improved in reference with the best forward flight blade. However, due to the smaller blade tip area in contrast to the baseline blade, the blade also rapidly loses its efficiency at higher thrust levels and is prone to stall earlier than the baseline blade. In the forward flight envelope, Figure 4.41, the blade has the biggest advantage at advance ratios between $\mu = 0.17 \ldots 0.27$, which is lessened at higher speeds. Yet, this blade remains superior to the baseline blade.

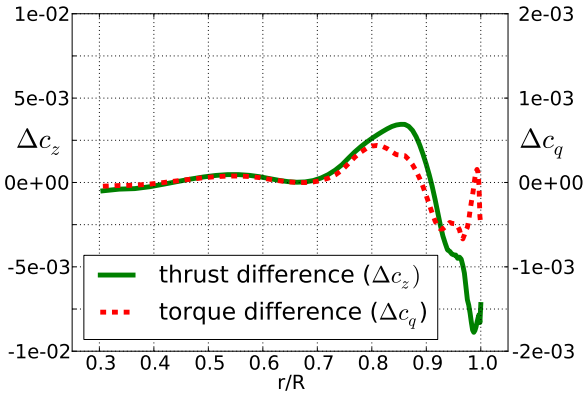
The second **trade-off** blade (TO) is pictured in Figure 4.38(b) features a good balance of both goal functions. It has more correlation with so far considered hover blades. The twist is larger over the whole blade span, except for the bump at r/R = 80 %. The tip sweep and dihedral increased further in relation to the TOF blade, adjusting the blade more towards hover. The inboard forward sweep is small in contrast to the hover blades, which poses an attribute of forward flight blades as it causes less of an elastic divergence. With the larger twist, the blade starts to exploit the previous tip vortex more than the TOF blade. This also enhances the behavior at the lower thrust regions, where it outperforms the baseline blade in the design condition and is slightly better than the baseline blade at higher thrust settings, Figure 4.40. In the forward flight regime, Figure 4.41, it is very similar to the baseline blade at lower advance ratios $\mu = 0.07 \ldots 0.16$, thus no gain in the rate of climb is expected. At higher advance ratios around $\mu = 0.23 \ldots 0.32$, it is superior over the baseline blade. This effect is lessened for advance ratios above 0.32.

The last **trade-off** blade is more biased towards **hover** (TOH) than the previous blade, Figure 4.38(c). The forward and backward sweep is larger than for the best hover blade. The waviness of the twist distribution is not observed as this blade focuses more on alleviating the tip vortex, rather than the root vortex. With the greater dihedral, it positions the blade better into the tip vortex in hover and allows it to raise the exploitation effect, Figure 4.39(f). It now supersedes the baseline blade in the whole hover envelope, but is not as superior in the higher thrust regions than the best hover blade. However, at lower and intermediate advance ratios around $\mu = 0.07 \ldots 0.23$ this blade requires more power than the baseline blade. This blade is therefore less suited for loitering and will reduce the maximum climb speed deduced from the polar in Figure 4.41.

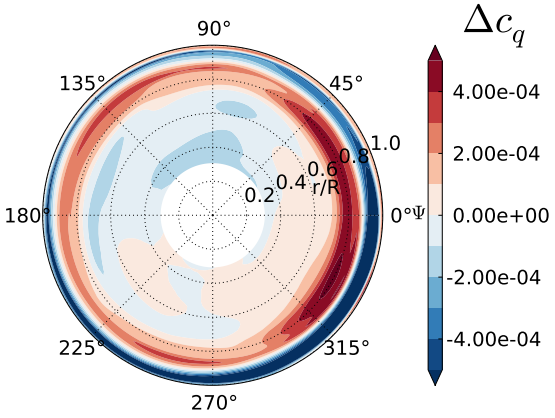
From the analysis of these various rotor configurations, the viability of the multi-objective approach is demonstrated. A fortunate result is that hover blades improved



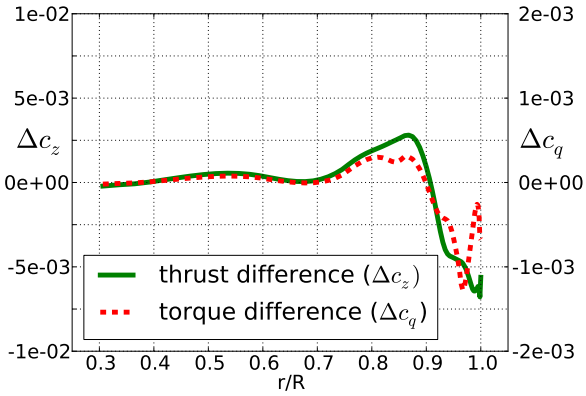
(a) TOF forward flight torque differences



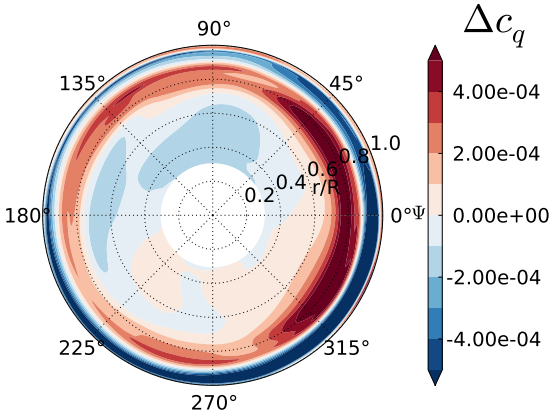
(b) TOF hover load differences



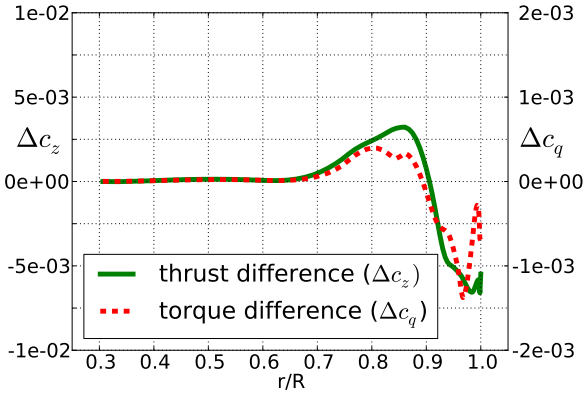
(c) TO forward flight torque differences



(d) TO hover load differences

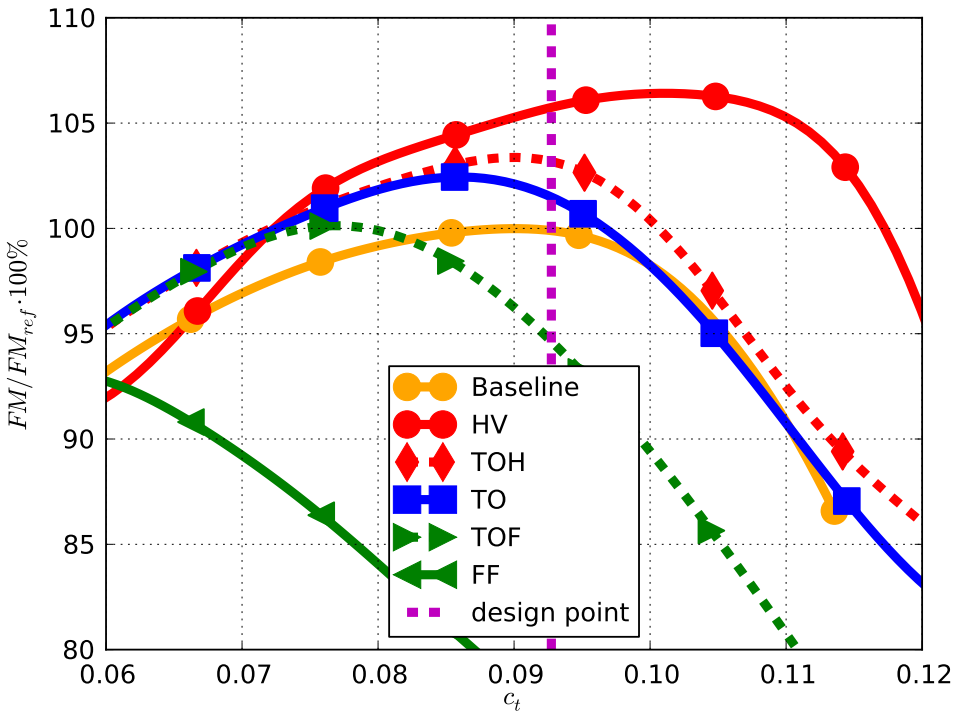


(e) TOH load differences



(f) TOH hover load differences

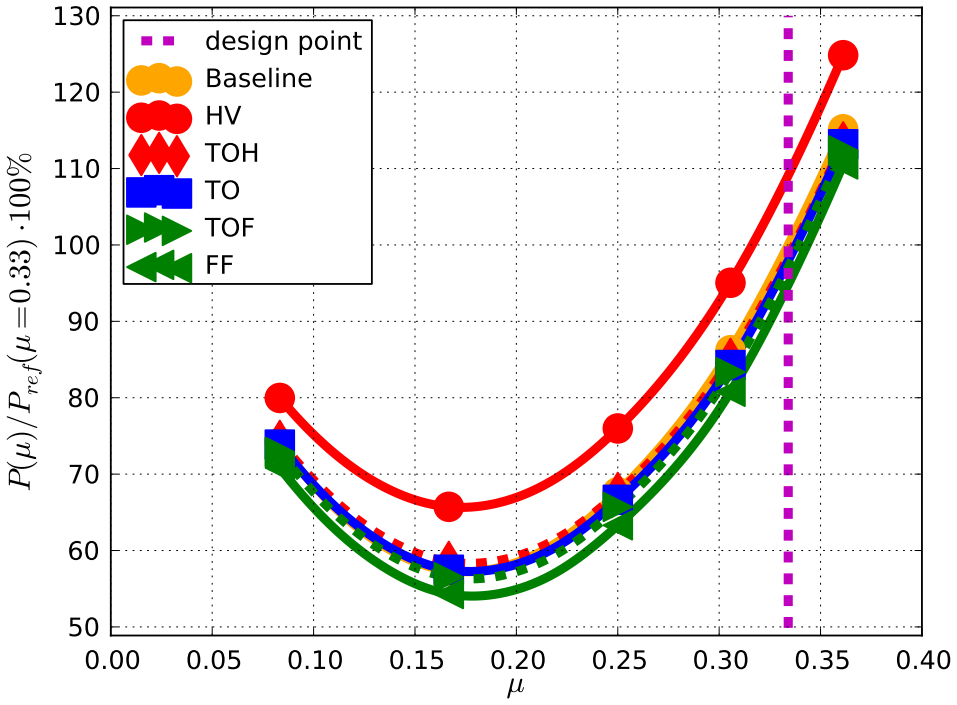
Figure 4.39: Load differences of selected GRC blades. (values of each blade minus baseline)



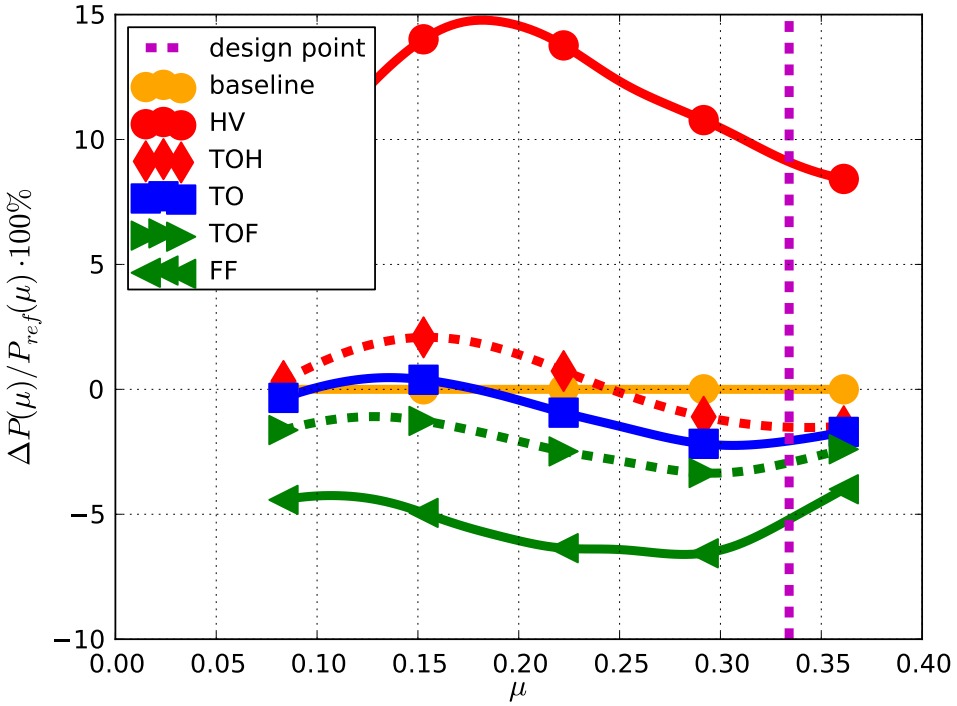
(a) **FNS** optimized blades

Figure 4.40: Performance polars of optimized GRC blades in hover

over the whole thrust polar, while forward flight blades over the whole speed regime. Caution is advised though as to generalize this fact. It is strongly dependent on the initial reference geometry as well as the particularly chosen design conditions.



(a) Absolute values. Values scaled by the performance of the reference rotor in the reference flight condition.



(b) Relative differences. Values scaled by the performance of the reference rotor at the respective velocity.

Figure 4.41: Performance polars of optimized GRC blades in forward flight

4.3 Reflection of the Variable-Fidelity Optimizations and Recommendations

Summing up this chapter, the variable-fidelity proved to be beneficial for the aerodynamic optimization of helicopter rotor blades. In Table 4.4, the cost savings when applying the variable-fidelity optimization strategy in contrast to the single-fidelity optimization strategy are listed for the high-fidelity optimizations of the 7A and GRC rotor blades. For this comparison, the best single- to variable-fidelity runs are taken. Reaching the goal function value within a specified threshold is better accelerated in most cases with the variable-fidelity than reaching the design or Pareto confidence. The reason for this is that per definition three more high-fidelity samples are required while also a globally more accurate surrogate model is required to confirm that the found optimum or Pareto front is valid. The greatest gain is noticed for reaching the threshold for the single-objective optimization of the GRC rotor with about 85 % cost savings for each. Here, the design space shrinking mechanism is additionally applied, which reduces the space to be filled with high-fidelity samples for the variable-fidelity approach. The least speed up is seen for the hover optimization of the 7A rotor with only 42 % cost savings. Here, the design space shrinking has not been utilized. The cost savings for reaching the design or Pareto confidence range between 55 % and 70 % for the presented cases.

	7A rotor			**GRC rotor**		
flight condition(s)	**hover**	**forward flight**	**both**	**hover**	**forward flight**	**both**
threshold	42 %	83 %	49 %	85 %	85 %	69 % **
design confidence	65 %	59 %	62 %	55 % *	70 %	N/A**

Table 4.4: Cost savings with variable-fidelity optimizations for either reaching the optimum (with a given threshold) or the design/Pareto confidence.
* - the GRC single-fidelity optimization at high-fidelity did not reach the design confidence, the value is based upon assumption that within the next three iterations it is reached.
** - For the multi-objective optimization of the GRC rotor, neither single- nor variable-fidelity optimization reached the Pareto confidence. The single-fidelity optimization featured one third the samples the variable-fidelity did with the same resource consumption.

Moving over to the observations made for the aerodynamic models that have been selected in Chapter 3. These specified fidelity levels prove their applicability for the 7A rotor optimization with four variables. Yet, when increasing the number of parameters of the optimization and further including constraints in the form of pitch link loads for the GRC rotor, a better physical agreement is required to capture the influence of viscosity as well as the blade tip vortex. The chosen low-fidelity in hover, the finite state inflow model **FISUW**, is not able to speed up the mid- or high-fidelity simulations as it lacks the modelling of the tip vortex, which turns out to be crucial for the hover flight condition with a finer parameterization. In forward flight, the chosen mid-fidelity for forward flight, the single-blade Euler simulation **EU**, is inferior to the chosen low-fidelity model, the blade element theory with the prescribe wake

model **P.WAKE**. The reason is that the GRC rotor is operated at a lower advance ratio and has five instead of four blades in contrast to the 7A rotor. This brings a greater vortex-vortex and blade-vortex interaction and thus neglecting this effect grants false rotor designs. However, this is not the only effect, as a second mid-fidelity is introduced for the GRC rotor, a multi-blade Euler simulation **MEU**. This method improves the overall rotor design, but is still not on par with the low-fidelity model in forward flight as it lacks viscosity. This and the increased cost for **EU** and **MEU** in comparison with **P.WAKE** further decrease the benefit for the variable-fidelity process. With these observations in mind, it is recommended when performing variable-fidelity optimizations for helicopter rotor blades, to utilize the inviscid Euler computations in hover and the blade element theory with a prescribed wake model in forward flight as the low-fidelity methods. The final recommended combination with the high-fidelity based on RANS in summarized in Table 4.5.

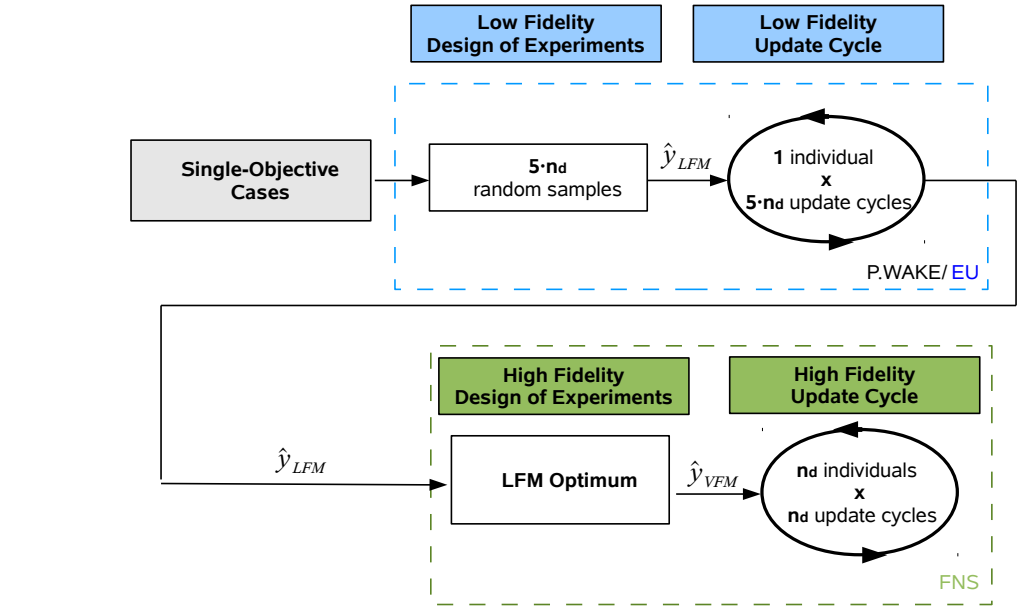
Fidelity	**hover**	**forward flight**
low	**EU**	**P.WAKE**
high	**FNS**	**FNS**

Table 4.5: Final selection of methods for the variable-fidelity approach

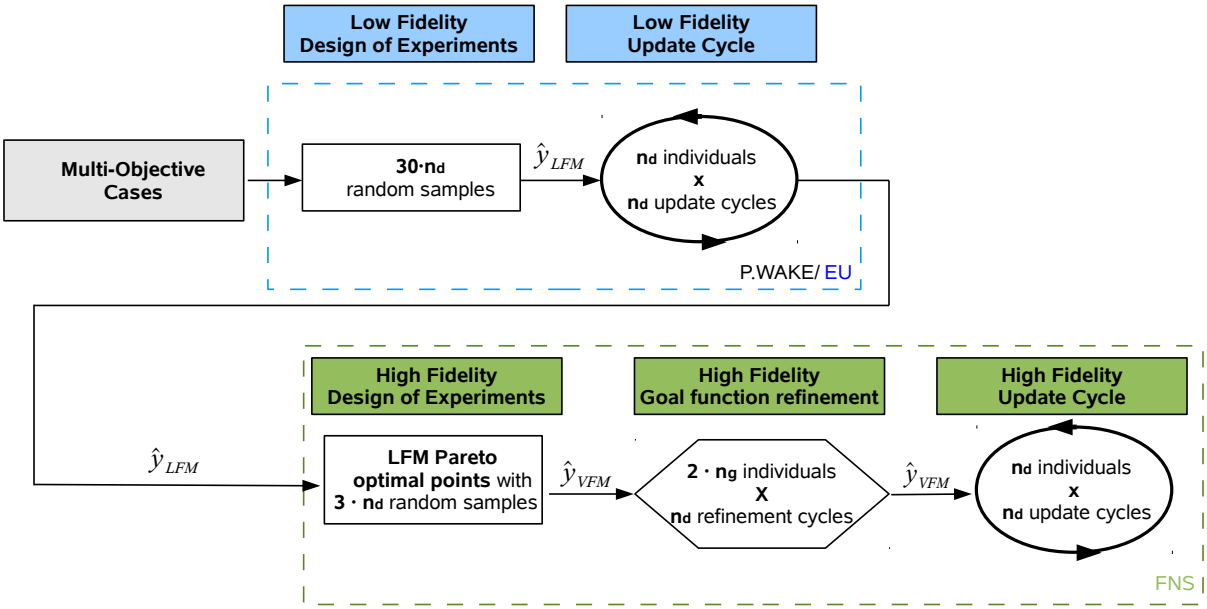
From the here investigated test cases, best practice recommendations are given for the aerodynamic planform optimization of helicopter rotor blades using variable-fidelity methods. These recommendations are made depending on the number of design parameter n_d applied in the optimization and the number of goal functions n_g.

For single-objective optimizations, it is proposed to apply five times the number of parameters to the size of the design of experiments for the low-fidelity and then grant the low-fidelity optimization cycle the same amount of samples, which will likely not be all used up. The high-fidelity design of experiments should then consist of the low-fidelity optimum along with a random design of experiments containing roughly n_d samples. The update cycle should then be allowed to sample ten times the number of parameters for reaching the design confidence. Shrinking the design space is also suggested. This guideline is visualized in the flowchart of Figure 4.42(a).

On the multi-objective side, the low-fidelity design of experiments should be sampled more generously with about ten to thirty times of n_d with an multi-objective update cycle of the length n_d with n_d samples each. The high-fidelity should then be a blend of points from the low-fidelity Pareto front and random samples containing about three times n_d. The high-fidelity surrogate models should be further refined in each goal function by sampling the best goal function value and expected improvement of the individual surrogate model. This should be repeated n_d times. Finally, the Pareto front is resolved by searching it with n_d update cycles with n_d samples each, the same way as for the low-fidelity. The recommendation for the multi-objective scenario is sketched in Figure 4.42(b).



(a) Single-objective cases



(b) Multi-objective cases

Figure 4.42: Recommended setups for variable-fidelity planform optimizations of helicopter rotor blades

5 Summary

The motivation of this thesis is to speed up the existing high-fidelity optimization framework for helicopter rotor blades created by Imiela [125]. His framework features high-fidelity CFD methods to compute the aerodynamics on a helicopter rotor coupled with a surrogate-based optimization algorithm. Acting on the idea of Collins [85], the chosen way for speeding up the optimization is to apply variable-fidelity based surrogate models. In contrast to Collins, a thorough quantification of the possible speed-up of the variable-fidelity process is done here. Also Collins' question of what are suitable combinations of simulations with different levels of fidelity for the rotor optimization process is answered.

Presented in Chapter 2, a novel surrogate based optimization framework for variable-fidelities is established. This framework contains state-of-the-art gradient-free optimization algorithms, design of experiments techniques as well as Kriging surrogate models, which ensures a correct comparison of single- to variable-fidelity optimization strategies. The chosen variable-fidelity model is based on Hierarchical Kriging developed by Han and Görtz [105], which also allows the handling of multiple fidelity levels. Their model is adjusted in this work by adding linear terms depending on the design parameters to grant more flexibility in case the low-fidelity model does not match exactly with the high-fidelity model. This surrogate model then features a higher accuracy than the response surface based bridging employed by Collins and greater robustness than the Co-Kriging models utilized by Yamazaki and Mavripilis [154] or Forrester et al. [78]. Additionally, two mechanisms to further accelerate the variable-fidelity optimization process are developed. The first mechanism is to exploit the knowledge of the low-fidelity method to construct a high-fidelity design of experiments by adding the low-fidelity optimum and samples nearby to the high-fidelity design of experiments. The second mechanism is similar to the technique by Li [93], where the design space is reduced. However, where Li sets the design space shrinkage based upon the number of samples evaluated and drops all samples outside the shrunk design space, the procedure here reduces the design space based upon the results of the low-fidelity models. The assumption here is that the high-fidelity optimum is in vicinity of the low-fidelity one, which is similar to the assumptions of Leusink et al. [127]. They initiate their genetic algorithm with the Pareto front generated from a low-fidelity method to continue the algorithm with a high-fidelity method.

The here established framework is then tested with a synthetic goal function to investigate the variable-fidelity optimization process. From this synthetic goal function it is seen that it is essential to pick a valid low-fidelity method as otherwise the variable-fidelity optimization is slowed down and might even be slower than a pure single-high-fidelity optimization. Therefore in Chapter 3, the abilities of available aerodynamic models for helicopter rotors are reviewed. On the one hand, the theory of the physics of the different aerodynamic models available is discussed along with the

requirements for an aerodynamic planform optimization, while on the other hand a practical investigation with these methods is performed. The different fidelity methods are compared in a parametric study of the planform parameters anhedral, chord, sweep and twist at the two major flight conditions of helicopter rotors, hover and forward flight. A genetic optimization containing all previously examined parameters is performed afterwards. From this study multiple conclusions are drawn. First, the individual parameters couple when all included in one optimization, which shifts the optimum from the one that would be estimated from the single parameter studies. Second, no single shape can be derived that satisfies both flight conditions equally well. An optimal rotor in the sense that both, hover and forward flight are ultimately satisfied does not exist and a good multi-objective approach is required to show the designer the potential trade-offs faced. Thirdly, simulation methods that performed well in hover do not necessarily perform well in forward flight and vice versa. This arises from the fact that different physical aspects become more important in the respective flight condition. For example the vortex dominated flow field in hover is quickly convected in forward flight, where fluid-structural coupling as well as transonic effects become more influential.

After this theoretical review and the numerical experiments of the aerodynamic models a potential set of low-, mid- and high-fidelity models is selected for each flight condition to be examined further.

In hover, the low-fidelity model is based on a finite-state inflow model, the mid-fidelity on inviscid Euler simulations on a coarse mesh, and the high-fidelity on a well resolved RANS solution. The finite-state model is preferred over wake models as the chosen low-fidelity, as the wake models show an exaggerated behavior for the design parameters and even predict the required power to be lower than the thermodynamically possible minimum. The finite state model however features no tip vortex in its solution and therefore does not influence the anhedral parameter. The inviscid Euler simulation is modeled as a periodically steady solution. In contrast to an equivalent RANS mesh, it grants more promising results, as less numerical damping is required for a stable solution. Thus the tip vortices are kept longer in the simulation and the resulting rotor configurations from the genetic optimization are close to the configurations generated with a well resolved RANS mesh. The latter is then the set high-fidelity simulation as it represents the state-of-the-art method for the hover flight condition.

In forward flight, a prescribed wake model is the chosen low-fidelity method, a single-blade inviscid simulation the mid-fidelity model and a multi-blade simulation including viscosity the high-fidelity model. The prescribed wake model resolves all physical effects which the high-fidelity also resolves, yet not on a first principles basis. Compressibility and viscous effects are incorporated through airfoil look-up tables and the tip vortices are modelled by inviscid potentials. The single-blade inviscid simulation neither features viscosity nor is the influence of the tip vortices from the other blades. However, it is able to model three dimensional effects in the near field of the blade and therefore shows promising results in the fast forward flight condition.

With these fidelity levels defined, two rotor configurations are chosen to test the variable-fidelity optimization approach in Chapter 4. First four design parameters are applied for improving the 7A model rotor. Second, in a more complex industrially relevant scenario comprising ten design variables and additional constraints, a specific blade from the European CleanSky Green RotorCraft (GRC) research program is optimized. The 7A optimizations are more focused on the ideal surrogate based settings concern-

ing optimal size of the design of experiments as well as combinations of low-, mid- and high-fidelity samples, while with the GRC optimizations the best variable-fidelity combinations as well as limitations to the variable-fidelity approach are found.

The rotor optimizations revealed that independently of the optimization mode, single- or variable-fidelity, an adequate size for the design of experiments is crucial for a successful as well as accurate optimization. It is observed that an optimal size exists for a specific problem to efficiently reach the optimum. A too small design of experiments will stall the optimization prematurely, while a too large design of experiments will waste resources. The optimal size for the design of experiments is reduced with the variable-fidelity approach from which the speed up of the optimization primarily arises. The reason is that the initial trend to start out the update cycle of the surrogate based optimization requires most features of the goal function landscape to be resolved for safely finding the optimum. Unfortunately, the size for the design of experiments cannot be explicitly set apriori. The rule of thumb proposed by Jones et al. [89] with ten samples per parameter is on the safe side when approximately half the number of samples is granted for the update cycle using the single-fidelity approach. Roughly one third in terms of high-fidelity samples is required for the variable-fidelity design of experiments, if the knowledge from the low-fidelity surrogate model is exploited for the selection of points for the high-fidelity design of experiments. Additionally reducing the design space based upon the low-fidelity optimum grants a variable-fidelity optimization about twice as fast as the variable-fidelity optimization without the shrinkage.

The investigation of dual- and triple-fidelities in the variable-fidelity process showed little difference in terms of optimization efficiency. This arises from the fact that either the low-fidelity yields noticeable different results and therefore does not accelerate the process, or the low-fidelity is already that good that the mid-fidelity can be skipped. Therefore dual-fidelity is advised for the sake of simplicity, robustness and speed. A greater impact on the efficiency is seen when surrogate models of low- or mid-fidelity simulations, which are abundantly randomly sampled, are replaced with surrogate models of surrogate based optimizations at low- or mid-fidelity level. Under the assumption that the low-fidelity optimum is close to the high-fidelity optimum, the refinement of this area in the low-fidelity is more beneficial than a globally resolved low-fidelity surrogate model. The overall turnaround time for the variable-fidelity optimization is then reduced as the cost of the low-fidelity surrogate model creation is reduced.

The ability to exploit the variable-fidelity approach for solving multi-objective problems is also analyzed. A Pareto based approach is selected for this examination, which demonstrates on the one hand the necessity of the multi-objective approach for reasonable rotor designs. On the other hand the variable-fidelity approach proves to be mandatory for it, due to the tremendous cost associated with multi-objective optimizations. Embedding the multi-objective optimization in a surrogate based approach costs roughly three times more samples for two objectives than the single-objective surrogate based optimization. For the complex multi-objective optimization with ten design parameters the variable-fidelity approach found about three times more viable samples for a similar amount of resources in comparison with an equivalent single-fidelity optimization. Posed differently, the additional cost for performing a high-fidelity multi-objective optimization with two goal functions instead of a single-objective optimization with only one goal function is compensated for by the variable-fidelity approach.

Another observation made during the preparation of this thesis is that for higher dimensional problems, where the impact of the individual parameters are of noticeable different magnitude, a clear mapping between the goal function values on the Pareto front and the according design parameters is difficult to make. Especially in shallow regions of either goal function, constructing a highly accurate surrogate model is costly and at the same time the optimization process struggles in identifying the path of the parameters through this landscape. Similar to the findings by Toal et al. [130], the importance of appropriately tuning the surrogate models is equally important as selecting the right search algorithm to find the optimum. No difference is observed between variable- and single-fidelity for this fact. The observation made by Toal et al. that those difficulties increase with higher dimensions is also underlined by this thesis.

From the applied variable-fidelity optimization of the GRC rotor, the most suitable combination for the aerodynamic planform optimization of helicopter rotor blades is abstracted. The low-fidelity consists of inviscid Euler simulation on a periodic mesh in hover and the blade element theory with a prescribed wake model in forward flight. The inviscid Euler simulations are necessary, as for the more detailed parameterization of the GRC rotor, the effect of vorticity becomes more important and thus the finite state inflow model clearly loses its validity. In forward flight, the prescribed wake model arrived at a superior rotor design in contrast to the inviscid single-blade computation. The tip vortices become also more important in forward flight for the improved parameterization and the neglection of friction leads to false fluid-structural coupling. Both effects, vorticity and friction are included in the blade element theory with a prescribed wake model, yet on a simplified model. The blade element theory with a prescribed wake model in forward flight and the inviscid CFD solution in hover are then combined with high-fidelity simulations based on the RANS equations. In hover as well in forward flight, sufficiently well resolved meshes are utilized, which then already pose as the re-computations Imiela does for his final configurations.

In terms of rotor configurations, a not yet named physical effect is discovered in hover. Here coined as vortex exploit, the optimizer tries to regain energy from the previous vortex by placing the blade tip into the vortex in such a way that the outboard part of the rotor generates thrust in the direction of rotation, thus decreasing the overall required power. This is achieved by a very high twist gradient towards the tip, which then tilts the resulting aerodynamic force of the affected airfoils. The an-/dihedral parameter then enhances this effect by placing the blade more optimal to the vortex. Such an effect has not been observed in forward flight.

The most important parameter for an aerodynamic optimization is the twist parameter, which unfortunately also comes out very differently for the two considered flight conditions, hover and forward flight. The two main drivers are the downwash and the vorticity field with a minor but not negligible influence from the structural coupling with other parameters. The chord distribution is driven by the trade-off of improving the circulation and reducing friction drag. The blade sweep is found to be similar for both flight conditions and is more conservative for forward flight than in hover. The effect Imiela [125] stated that sweep locally modifies the twist through fluid-structural coupling is rephrased for higher dimensional problems. The twist of a swept blade is adjusted in such a way that the occurring torsion is countered. The off-loading effect of anhedral, which is employed on most current rotor designs, is only observed for very specific conditions. Both in hover and forward flight, a different setting is chosen which results in a dihedral of different magnitude at each flight condition. While in

forward flight, the thrust distribution is more evenly re-organized on the rotor disc, in hover the dihedral places the tip vortex in a more suitable way for its exploitation.

Wrapping it up, this thesis proved that utilizing variable-fidelity models significantly reduces the required resources to perform an aerodynamic planform optimization for helicopter rotor blades. For the complex test case with ten design parameters, 85 % cost savings are achieved in hover as well as forward flight and 325 % more Pareto optimal samples are found at comparable resources for the multi-objective optimization. It also becomes clear that for a good blade design the multi-objective approach is indispensable and it only becomes affordable with the variable-fidelity approach. At the same time, the most advantageous combination of aerodynamic models is determined. The conditions and model combinations under which this approach becomes inefficient have been discussed. Modifications to the surrogate based optimization process using variable-fidelity are made as needed for the rotor optimization. A byproduct of this work is a set of novel rotor designs.

As a future outlook, the here developed methodology should be applied for multi-disciplinary rotor design instead of pure aerodynamic optimizations. In particular an enhancement in the structural modeling, such as directly coupling the computational fluid dynamics simulation with a structural finite element code is recommended. This will also allow performing structural optimizations, where the eigenmodes of the blades are better aligned with their operational frequencies. Adding aero-acoustics into the consideration as well as other off-design conditions or even perform complete helicopter simulations will allow the future blade designer to create new generations of rotor blades.

Bibliography

[1] G. Wilke: Variable Fidelity Optimization of Required Power of Rotor Blades: Investigation of Aerodynamic Models and their Application. In: *38th European Rotorcraft Forum*, 2012

[2] G. Wilke: Multi-Objective Optimizations in Rotor Aerodynamics using Variable Fidelity Simulations. In: *39th European Rotorcraft Forum*, 2013

[3] M. Imiela, and G. Wilke: Passive Blade Optimization and Evaluation in Off-Design Conditions. In: *39th European Rotorcraft Forum*, 2013

[4] G. Wilke: Applying Multi-Objective Variable-Fidelity Optimization Techniques to Industrial Scale Rotors: Blade Designs for CleanSky. In: *41th European Rotorcraft Forum*, 2015

[5] R. Celi: Recent Applications Of Design Optimization To Rotorcraft - A Survey. In: *Journal of Aircraft* 36 (1999), S. 176–189

[6] R. Ganguli: Survey of Recent Developments in Rotorcraft Design Optimization. In: *Journal of Aircraft* 41, No. 3 (2004), S. 493–510

[7] A. Brocklehurst, and G. N. Barakos: A review of helicopter rotor blade tip shapes. In: *Progress in Aerospace Sciences* 56 (2013), Nr. 0, 35 - 74. – ISSN 0376–0421

[8] M. Bebesel, G. Polz, and E. Schöll: Aerodynamic and Aeroacoustic Layout of the ATR (Advanced Technology Rotor). In: *55th Annual Forum of the American Helicopters Society*, 1999

[9] M. Hollands, M. Kessler, and E. Krämer: Influence of An-/Dihedral and of Different Blade-Shapes on Performance and Aeroacoustics of an Isolated Rotor. In: *38th European Rotorcraft Forum*, 2012

[10] J. D. Baeder: Passive Design for Reduction of High-Speed Impulsive Rotor Noise. In: *52nd Annual Forum of the American Helicopters Society*, 1996

[11] H. E. Jones, and C. L. Burley: A Study of the Effects of Blade Shape on Rotor Noise. In: *AHS Technical Specialists Meeting for Rotorcraft Acoustics and Aerodynamics*, 1997

[12] W. G. Bousman: Airfoil Design and Rotorcraft Performance. In: *58th Annual Forum of the American Helicopters Society*, 2002

[13] S. Schneider: *Optimierung von modernen Hubschrauberrotoren mit CFD*, Universitt Stuttgart, Mastersthesis, 2004

[14] J.G. Leishman, and K. M. Rosen: Challenges in the Aerodynamic Optimization of High-Efficiency Proprotors. In: *Journal of the American Helicopter Society* 56 (2011), S. 012004–1–20

[15] S. Bhadra, and R. Ganguli: Aeroelastic Optimization of a Helicopter Rotor: Using Orthogonal Array—Based Metamodels. In: *AIAA Journal* 44 (2006), S. 1941–1951

[16] B. Glaz, P. P. Friedmann, and L. Liu: Surrogate Based Optimization of Helicopter Rotor Blades for Vibration Reduction in Forward Flight. In: *47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2006

[17] J. K. Tritschler, M.Syal, R. Celi, and J. G. Leishman: A Methodology for Rotorcraft Brownout Mitigation: Using Rotor Design Optimization. In: *American Helicopter Society 66th Forum*, 2010

[18] K. K. Saijal, R. Ganguli, and S. R. Viswamurthy: Optimization of Helicopter Rotor Using Polynomial and Neural Network Metamodels. In: *Journal of Aircraft* 48-2 (2011), S. 553–566

[19] K. Mani, B. A. Lockwood, and D. J. Mavriplis: Adjoint-based Unsteady Airfoil Design Optimization with Application to Dynamic Stall. In: *68th Annual Forum of the American Helicopters Society*, 2012

[20] V. Ahuja, C. Kannepalli, A Zambon, H. Lapin, and Mark Potsdam: Unsteady Blade Shape Optimization for Rotorcraft. In: *70th Annual Forum of the American Helicopters Society*, 2014

[21] B. R. Jones, W. A. Crossley, and A. S. Lyrintzis: Aerodynamic and Aeroacoustic Optimization of Rotorcraft: Airfoils via a Parallel Genetic Algorithm. In: *Journal of Aircraft* 37.6 (2000), S. 1088–1096

[22] A.T. Conlisk: Modern helicopter rotor aerodynamics. In: *Progress in Aerospace Sciences* 37 (2001), Nr. 5, S. 419 – 476. – ISSN 0376–0421

[23] *Kapitel* On the Mechnaical Principles of the Action of Propellers. In: W. J. M. Rankine: *Transactions of the Institute of Naval Architects*. 1865, S. 13–39

[24] *Kapitel* On the Part Played in Propulsion by Differences in Fluid Pressure. In: R.E. Froude: *Transactions of the Institute of Naval Architects*. 1889, S. 390

[25] *Kapitel* Airplane Propellers. In: H. Glauert: *Aerodynamic Theory*. Springer Berlin / Heidelberg, 1935, S. L

[26] S. Drzewieki: *Thèorie Gènèrale de l'Hèlice*. Paris, 1920

[27] A. Gessow, and G. C. Myers, Jr.: *Aerodynamics of The Helicopter*. Ungar, 1983

[28] Leishman, J.G.: *Principles of Helicopter Aerodynamics*. Cambridge University Press, 2006

[29] D. L. Kunz: Comprehensive Rotorcraft Analysis: Past, Present, and Future. In: *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference*, 2005

[30] W. Johnson: Rotorcraft Aerodynamics Models for a Comprehensive Analysis. In: *54th Annual Forum of the American Helicopters Society*, 1998

[31] G. Arnaud, B. Benoit, and F. Toulmay: Improvements to the Aerodynamic Model of the R85 Helicopter Rotor Code: Validation and Applications. In: *28th ISL Applied Aerodynamics Symposium*, 1991

[32] B. Benoit, A.-M. Dequin, K. Kampa, W. von Grünhagen, P.-M. Basset, and B. Gimonet: HOST, a General Helicopter Simulation Tool for Germany and France. In: *56th Annual Forum of the American Helicopters Society*, 2000

[33] D.M. Pitt, and D.A. Peters: Theoretical Prediction of Dynamic Inflow Derivatives. In: *Vertica* 5 (1981), S. 21–34

[34] W. Kinner: Die Kreisformige Tragfläche auf Potentialtheorischer Grundlage / Ingenieur Archiv, vol. 7. 1937. – Forschungsbericht

[35] D. A. Peters, D. B. Boyd, and C. J. He: Finite-State Induced-Flow Model for Rotors in Hover and Forward Flight. In: *43rd Annual Forum of the American Helicopter Society*, 1987

[36] P.-M. Basset, O. Heuze, J.V.R. Prasad, and M. Hamers: Finite State Rotor Induced Flow Model For Interferences and Ground Effect. In: *57th Annual Forum of the American Helicopters Society*, 2001

[37] T.A. Egolf, and A.J. Landgrebe: Helicopter Rotor Wake Geometry and Its Influence in Forward-Flight Volume I - Generalized Wake Geometry Wake Effect on Rotor Airloads and Performance / NASA Contractor Report 3726. 1983. – Forschungsbericht

[38] T. A. Egolf, and A. J. Landgrebe: Helicopter Rotor Wake Geometry and Its Influence in Forward Flight, Volume II - Wake Geometry Charts / NASA CR-3726. 1983. – Forschungsbericht

[39] T.S. Beddoes: A Wake Model for High Resolution Airloads. In: *Second International Conference on Basic Rotorcraft Research*, 1985

[40] G. Arnaud, and P. Beaumier: Validation of R85/Metar on the Puma RAE Flight Tests. In: *18th European Rotorcraft Forum*, 1992

[41] J. D. Kocurek, and J. L. Tangler: A Prescribed Wake Lifting Surface Hover Performance Analysis. In: *32nd Annual National Forum of the American Helicopter Society*, 1976

[42] D. R. Clark, and A. C. Leiper: THE FREE WAKE ANALYSIS - A method for the prediction of helicopter rotor hovering performance. In: *25th Annual National Forum American Helicopter Society*, 1969

[43] S.G. Sadler: A method for predicting helicopter wake geometry, wake-induced flow and wake effects on blade airloads. In: *27th Annual Forum of the American Helicopter Society*, 1971

[44] T.A. Egolf: Helicopter Free Wake Prediction of Complex Wake Structures Under Blade-Vortex Interaction Operating Conditions. In: *44th Annual Forum of the American Helicopter Society*, 1988

[45] B. Michea: *Etude des Sillages de Rotors D'Helicoptere en Vol D'Avancement et de leur Influence sur les Performances du Rotor. (Interaction Pale-Tourbillon)*, Universite Paris 6, Dissertation, 1992

[46] P. Beaumier, and Y. Delrieux: Description and validation of the ONERA computational method for the prediction of blade-vortex interaction noise. In: *Aerospace Science and Technology* 9 (2005), Nr. 1, 31 - 43. – ISSN 1270–9638

[47] B. G. van der Wall, S. N. Jung, J. W. Lim, M. J. Smith, J. Bailly, J. D. Baeder, and D. D. Boyd, Jr.: An Assessment of Comprehensive Code Prediction State-of-the-Art Using the HART II International Workshop Data. In: *Annual Forum 68 of the American Helicopters Society*, 2012

[48] S. R. Ahmed, and V. T. Vidjaja: Unsteady Panel Method Calculation of Pressure Distribution on BO 105 Model Rotor Blades and Validation with DNW-Test Data. In: *50th Annual Forum of the American Helicopters Society*, 1994

[49] J. Yin: Main Rotor and Tail Rotor Blade Vortex Interaction Noise under the Influence of the Fuselage. In: *38th European Rotorcraft Forum*, 2012

[50] A. Massaro, A. D'Andrea, and E. Benini: Multiobjective-Multipoint Rotor Blade Optimization in Forward Flight Conditions Using Surrogate-Assisted Memetic Algorithms. In: *37th European Rotorcraft Forum*, 2011

[51] A. Massaro, and A. D'Andrea: Multi-Point Aerodynamic Optimization by Means of Memetic Algorithm for Design of Advanced Tiltrotor Blades. In: *39th European Rotorcraft Forum*, 2013

[52] R. C. Strawn, F. X. Caradonna, and E. P. N. Duque: 30 Years of Rotorcraft Computational Fluid Dynamics Research and Development. In: *Journal of the American Helicopter Society* (2005)

[53] F. X. Caradonna, and M. P. Isom: Subsonic and Transonic Potential Flow over Helicopter Rotor Blades. In: *AIAA 10th Aerospace and Sciene Meeting*, 1972

[54] T.A. Egolf, and S.P. Sparks: A Full Potential Rotor Analysis with Wake Influence Using and Inner-Outer Domain Technique. In: *42nd Annual Forum of the American Helicopter Society*, 1986

[55] P. Beaumier, J. Zibi, and M. Costes: CFD Drag and Power Prediction for a Rotor in Hover or Forward Flight. Formulation and First Applications. In: *52nd Annual Forum of the American Helicopters Society*, 1996

[56] Norbert Kroll: *Berechnung von Strömungsfeldern um Propeller und Rotoren im Schwebeflug durch Lösung der Euler Gleichungen*, TU Braunschweig, Dissertation, 1989

[57] B.E. Wake, and L. N. Sankar: Solutions of Navier-Stokes Equations for the Flow About a Rotor Blade. In: *Journal of the American Helicopter Society* 34 (1989), Nr. 2, S. 13–23

[58] E. P. N. Duque, and G. R. Srinivasan: Numerical Simulation of a Hovering Rotor Using Embedded Grids. In: *48th Annual Forum of the American Helicopters Society*, 1992

[59] R. Stangl, and L. Wagner: Calculation of Steady Rotor Flow Using an Overlapping Embedded Grid Technique. In: *18th European Rotorcraft Forum*, 1992

[60] M. Amiraux, J. D. Baeder, and S. N. Koushik: Improved Correlation with HART-II Rotor Test Data Using Coupled CSD/CFD and Three Levels of Numerical Modelization. In: *38th European Rotorcraft Forum*, 2012

[61] K. Pahlke, and B. van der Wall: Progress in weak fluid-structure-coupling for multi-bladed rotors in high-speed forward flight. In: *28th European Rotorcraft Forum*, 2002

[62] J. Raddatz, and K. Pahlke: 3D Euler Calculations of Multibladed Rotors in Hover: Investigation of the Wake Capturing Properties. In: *AGARD Symposion on Aerodynamics and Aeroacoustsics of Rotorcraft*, 1994

[63] U. Kowarsch: Aeroacoustic Simulation of a Complete H145 Helicopter in Descent Flight. In: *41st European Rotorcraft Forum*, 2015

[64] M. Potsdam, H. Yeo, and R. Ormiston: Performance and Loads Predictions of a slowed UH-60A Rotor at High Advance Ratios. In: *39th European Rotorcraft Forum*, 2013

[65] R. E. Brown: Rotor Wake Modeling for Flight Dynamic Simulation of Helicopters. In: *AIAA Journal* (2000)

[66] G. R. Whitehouse, and H. Tadghighi: Investigation of Hybrid GridBased Computational Fluid Dynamics Methods for Rotorcraft Flow Analysis. In: *Journal of American Helicopter Society* 56 (2011), S. 032004–1–032004–9

[67] J. Steinhoff, and G. K. Raviprakash: Navier-Stokes computation of blade-vortex interaction using vorticity confinement. In: *AIAA paper* 95 (1995), S. 0161

[68] M. Costes: Development of a 3rd-order Vorticity Confinement Scheme for Rotor Wake Simulations. In: *38th European Rotorcraft Forum*, 2012

[69] M. J. Smith, R. Jain, A. Grubb, and K. Jacobson: Time-An-Spatially Adaptating Simulations for Efficient Dynamic Stall Predictions. In: *41st European Rotorcraft Forum*, 2015

[70] S. Choi, J. J. Alonso, E. v.d. Weide, and J. Sitaraman: Validation Study of Aerodynamic Analysis Tools for Design Optimization of Helicopter Rotors. In: *25th AIAA Applied Aerodynamics Conference*, 2007

[71] K. Kaufmann, M. Costes, F. Richez, A. D. Gardner, and A. Le Pape: Numerical investigation of three-dimensional dynamic stall on an oscillating finite wing. In: *70th Annual Forum of the American Helicopters Society*, 2014

[72] F. R. Menter: Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications. In: *AIAA-Journal* 32 (1994), S. 1598–1605

[73] M. Dietz, T. Kneisch, G. Roth, A. DAlascio, and D. Schimke: EC145 T2: Comprehensive and Challenging Industrial CFD Applications. In: *70th Annual Forum of the American Helicopters Society*, 2012

[74] M. Dietz, W. Khier, B. Knutzen, S. Wagner, and E. Krämer: Numerical Simulation of a Full Helicopter Configuration Using Weak Fluid-Structure Coupling. In: *American Institute of Aeronautics and Astronautics*, 2007

[75] W. Khier, M. Dietz, T. Schwarz, and S. Wagner: Trimmed CFD Simulation of a Complete Helicopter Configuration / Institut für Aerodynamik und Strömungstechnik, DLR, Braunschweig, Institut für Aerodynamik und Gasdynamik, Universität Stuttgart. 2008. – Forschungsbericht

[76] M. Dietz, and O. Dieterich: Towards Increased Industrial Application of Rotor Aeroelastic CFD. In: *35th European Rotorcraft Forum*, 2009

[77] M.D. Morris, and T. Mitchell: Exploratory designs for computer experiments. In: *Journal for Statistical Planning and Interference* 43 (1995), S. 381–402

[78] A. I. J. Forrester, A. Sòbester, and A. J. Keane: Multi-fidelity optimization via surrogate modelling. In: *The Royal Society* 463 (2007), S. 3251–3269

[79] L. Ju, Q. Du, and M. Gunzburger: Probabilistic methods for centroidal Voronoi tessellations and their parallel implementations. In: *Parallel Computing* 28 (2002), S. 1477–1500

[80] S. Lloyd: Least square quantization in PCM. In: *IEEE Transactions on Information Theory* 28 (1982), S. 129137

[81] J. MacQueen: Some methods for classication and analysis of multivariate observations. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* 1 (1967), S. 281297

[82] V. J. Romero, J. V. Burkardt, M. D. Gunzburger, and J. S. Peterson: Comparison of pure and "Latinized" centroidal Voronoi tessellation against various other statistical sampling methods. In: *Reliability Engineering & System Safety* 91 (2006), S. 1266–1280

[83] J. H. Halton: On the Efficiency of Certain Quasi-Radnom Sequences of Points in Evaluating Multi-Dimensional Integrals. In: *Numerical Mathematics* 2 (1960), S. 84–90

[84] J. M. Hammersley: Monte Carlo Methods for Solving Multivariate Problems. In: *Annals of the New York Academy of Science* 86 (1960), S. 844–874

[85] K. B. Collins: *A Multi-Fidelity Framework for Physics Based Rotor Blade Simulation and Optimization*, Georgia Institute of Technology, Dissertation, 2008

[86] C. Li, F.-L. Wang, Y.-Q. Chang, and Y. Liu: A modified global optimization method based on surrogate model and its application in packing profile optimization of injection molding process. In: *International Journal of Advanced Manufacturing Technology* 48 (2010), S. 506–511

[87] D. Krige: *A statistical approach to some mine valuation and allied problems on the Witwatersrand*, University of the Witwatersrand, Dissertation, 1951

[88] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn: Design and Analysis of Computer Experiments. In: *Statistical Science* 4-4 (1989), S. 409–435

[89] D. R. Jones, M. Schonlau, and W. J. Welch: Efficient Global Optimization of Expensive Black-Box Functions. In: *Journal of Global Optimization* 13 (1998), S. 455–492

[90] J. Knowles: ParEGO: A Hybrid Algorithm With On-Line Landscape Approximation for Expensive Multiobjective Optimization Problems. In: *IEEE Transactions on Evolutionary Computation* 10 (2005), S. 1

[91] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan: A fast and elitist multiobjective genetic algorithm: NSGA-II. In: *IEEE Transactions on Evolutionary Computation* 6 (2002), Nr. 2, S. 182 –197. – ISSN 1089–778X

[92] B. Horowitz, L. J. N. Guimarães, and S. M. B. Afonso: A Concurrent Efficient Global Optimization Algorithm: Applied to Engineering Problems. In: *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference 10 - 12 September 2008, Victoria, British Columbia Canada*, 2008

[93] C. Li: *A Surrogate-Based Framework with Hybrid Refinement Strategies for Aerodynamic Shape Optimization*, Technische Universität Braunschweig, Dissertation, 2013

[94] U. Siller, C. Vo, and E. Nicke: Automated Multidisciplinary Optimization of a Transonic Axial Compressor. In: *American Institute of Aeronautics and Astronautics*, 2007

[95] A. Forrester: *Efficient Global Aerodynamic Optimisation Using Expensive Computational Fluid Dynamics Simulations*, Faculty of Engineering, Science and Mathematics School of Engineering Sciences University of Southampton, Dissertation, 2004

[96] S. Jeong, M. Murayama, and K. Yamamoto: Efficient Optimization Design Method using Kriging Model. In: *Journal of Aircraft* 42-2 (2004), S. 413–420

[97] C. Johnson: *Optimisation of Aspects of Rotor Blades using Computational Fluid Dynamics*, University of Liverpool, Dissertation, 2012

[98] T. D. Robinson, K. E. Willcox, M. S. Eldred, and R. Haimes: Multifidelity Optimization for Variable-Complexity Design. In: *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Virginia, US*, 2006

[99] T. D. Robinson: *Surrogate-Based Optimization using Multidelity Models with Variable Parameterization*, Massachusetts Institute of Technology, Dissertation, 2007

[100] X. Cai, and F. Ladeinde: A Comparison of Two POD Methods for Airfoil Design Optimization. In: *35th AIAA Fluid Dynamics Conference and Exhibit 6 - 9 June 2005, Toronto, Ontario Canada*, 2005

[101] K. Carlberg, and C. Farhat: A Compact Proper Orthogonal Decomposition Basis for Optimization-Oriented Reduced-Order Models. In: *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2008

[102] J.Peter, and M. Marcelet: Comparison of surrogate models for turbomachinery design. In: *WSEAS Transactions on Fluid Mechanics* 3 (2008), S. 10–17

[103] S. E. Gano: *Simulation-Based Design Using Variable Fidelity Optimization*, University of Notre Dame, Dissertation, 2005

[104] W. Yamazaki, and D. J. Mavriplis: Derivative-Enhanced Variable Fidelity Surrogate Modeling for Aerodynamic Functions. In: *AIAA - Aerospace Sciences Meeting and Exhibit*, 2011

[105] Z. Han, S. Görtz, and R. Hain: A Variable-Fidelity Modeling Method for Aero-Loads Prediction. In: A. DILLMANN, G. HELLER, M. KLAAS, H-P. KREPLIN, W. NITSCHE, AND S. WOLFGANG (Hrsg.): *New Results in Numerical and Experimental Fluid Mechanics VII* Bd. 112. Springer Berlin / Heidelberg, 2010. – ISBN 978–3–642–14242–0, S. 17–25

[106] Z.-H. Han, and S. Görtz: A Hierarchical Kriging Model for Variable-Fidelity Surrogate Modeling. In: *AIAA Journal* 50-9 (2012), 1885-1896

[107] Y. Xiong, W. Chen, and K.-L. Tsui: A New Variable-Fidelity Optimization Framework Based on Model Fusion and Objective-Oriented Sequential Sampling. In: *Journal of Mechanical Design* 130 (2008), Nr. 11, 111401

[108] N. M. Alexandrov, J.E. Dennis, R.M. Lewis, and V. Torczon: A Trust Region Framework for Managing the Use of Approximation Models in Optimization. In: *In Proceedings of the Sixth AIAA/NASA/ISSMO Symposium on Multidisci- plinary Analysis and Optimization*, 1996

[109] R. Fletcher, and C. M. Reeves: Function Minimization by Conjugate Gradients. In: *Computer Journal* 6 (1964), S. 149–154

[110] M. S. Eldred, A. A. Giunta, S. F. Wojtkiewicz, Jr., and T. G. Trucano: Formulations for Surrogate-Based Optimization under Uncertainty. In: *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2002

[111] S. E. Gano, V. M. Perez, J. E. Renaud, and S. M. Batill: Multilevel Variable Fidelity Optimization of a Morphing Unmanned Aerial Vehicle. In: *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference*, 2004

[112] R. Hooke, and T. A.Jeeves: Direct Search Solution of Numerical and Statistical Problems. In: *Journal of Assosication of Computing Machinery* 8 (1961), April, 212–229. – ISSN 0004–5411

[113] J.A. Nelder, and R. Mead: A simplex function for minimization. In: *Computer Journal* 8-1 (1965), S. 308–313

[114] A. Le Pape: Numerical aerodynamic optimization of helicopter rotors: multi-objective optimization in hover and forward flight conditions / Applied Aerodynamics Department ONERA, BP 72, 92322 Châtillon Cedex, France. 2005. – Forschungsbericht

[115] S. Chae, K. Yee, C. Yang, T. Aoyama, S. Jeong, and S. Obayashi: Helicopter Rotor Shape Optimization for the Improvement of Aeroacoustic Performance in Hover. In: *Journal of Aircraft* 47-5 (2010), S. 1170–1783

[116] R. Storn, and K. Price: Differential Evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces. In: *Journal of Global Optimization* 11 (1997), S. 341–359

[117] S. W. Lee, and O. J. Kwon: Aerodynamic Shape Optimization of Hovering Rotor Blades in Transonic Flow Using Unstructured Meshes. In: *AIAA Journal* 44-8 (2006), S. 1816–1825

[118] A. Dumont, A. Le Pape, J. Peter, and S. Huberson: Aerodynamic Shape Optimization of Hovering Rotors Using a Discrete Adjoint of the Reynolds-Averaged Navier-Stokes Equations. In: *Journal of American Helicopter Society* 56 (2011), 032002-1-11

[119] F. Richez, and A. Dumont: Numerical optimizations of helicopter rotor twist law in hover and forward flight / ONERA, Applied Aerodynamics Department 8, rue des Vertugadins - 92190 Meudon, France. 2011. – Forschungsbericht

[120] S. Choi , M. Potsdam, K. Lee, G. Iaccarino, and J. J. Alonso: Helicopter Rotor Design Using a Time-Spectral and Adjoint-Based Method. In: *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference 10 - 12 September 2008, Victoria, British Columbia Canada*, 2008

[121] N. A. Vu, H.-J. Kang, A. I. Azamatov, J.-W. Lee, and Y.-H. Byun: Aerodynamic Design Optimization of Helicopter Rotor Blades in Hover Performance using Advanced Configuration Generation Method. In: *35th European Rotorcraft Forum*, 2009

[122] A. Le Pape, and P. Beaumier: Numerical optimization of helicopter rotor aerodynamic performance in hover. In: *Aerospace Science and Technology* 9 (2005), S. 191–201

[123] M. Imiela: High-fidelity optimization framework for helicopter rotors. In: *Aerospace Science and Technology* 23 (2012), Nr. 1, 2 - 16. – ISSN 1270–9638. – 35th ERF: Progress in Rotorcraft Research

[124] E. R. Leon, A. Le Pape, J-A. Desiderie, D. Alfano, and M. Costes: Concurrent Aerodynamic Optimization of Rotor Blades Using a Nash Game Method. In: *69th Annual Forum of the American Helicopters Society*, 2013

[125] M. Imiela: *Mehrpunktoptimierung eines Hubschrauberrotors im Schwebe- und Vorwärtsflug unter Berücksichtigung der Fluid-Struktur-Wechselwirkung*, Institut für Aerodynamik und Strömungstechnik, DLR, Braunschweig, Dissertation, 2012

[126] K. B. Collins, L. N. Sankar, and D. N. Mavris: Application of Low- and High-Fidelity Simulation Tools to Helicopter Rotor Blade Optimization. In: *Journal of the American Helicopter Society* 58 (2013), Nr. 042003, S. 1–10

[127] D. Leusink, D. Alfano, and P. Cinnella: Multi-fidelity optimization strategy for the industrial aerodynamic design of helicopter rotor blades. In: *Aerospace Science and Technology* 42 (2015), Nr. 0, 136 - 147. – ISSN 1270–9638

[128] A. Forrester, A. Sòbester, and A. Keane: *Engineering Design via Surrogate Modelling - A Practical Guide*. John Wiley & Sons Ltd., 2008

[129] D. Robinson, and C. Atcitty: Comparison of quasi-and pseudo-Monte Carlo sampling for reliability and uncertainty analysis. In: *Proceedings of the AIAA Probabilistic Methods Conference, St. Louis MO, AIAA99-1589*, 1999

[130] D. J. J. Toal, N. W. Bressloff, and A. J. Keane: Kriging Hyperparameter Tuning Strategies. In: *AIAA Journal* Vol. 46, No. 5 (2008), S. 1240–1252

[131] Z.-H. Han, R. Zimmermann, and S. Görtz: A New Cokriging Method for Variable-Fidelity Surrogate Modeling of Aerodynamic Data. In: *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, 2010

[132] H. Wackernagel: *Multivariate geostatistics: an introduction with applications*. Springer Science & Business Media, 2013

[133] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar: Differential Evolution Using a Neighborhood-Based Mutation Operator. In: *IEEE Transactions on Evolutionary Computation* 13-3 (2009), S. 526–

[134] R.W. Prouty: *Helicopter Performance, Stability and Control*. PWS Engineering Boston, 1986

[135] W. Bittner: *Flugmechanik der Hubschrauber*. Springer Berlin / Heidelberg, 2005

[136] A. Gessow: Effect of Rotor-Blade Twist and Plan-Form Taper on Helicopter Hovering Performance / NACA Technical Notice 1542. 1948. – Forschungsbericht

[137] G. Heilers: *Aerodynamische Mehrpunktoptimierung von Rotorblättern mit reibungsfreien Strömungssimulationen*, IAG Stuttgart, Masterstheis, 2014

[138] J. M. Drees: A Theory of Airflow Through Rotors and Its Application to Some Helicopter Problems. In: *Journal of the Helicopter Assoziation of Great Britain* 3 (1949), July-Sept, Nr. 2, S. 79–140

[139] C. He: *Development and application of a generalized dynamic wake theory for lifting rotors*, Georgia Institute of Technology, Dissertation, 1989

[140] J.P. Yin, and S.R. Ahmed: Treatment of Unsteady Rotor Aerodynamics Using a 3-D Panel Method / Deutsche Forschungsantalt für Luft- und Raumfahrt Forschungsbereich Strömungsmechanik. 1994. – Forschungsbericht

[141] J. Raddatz, and J. Fassbender: Block structured Navier-Stokes solver FLOWer. MEGAFLOW - Numerical Flow Simulation for Aircraft Design. In: *Notes on Numerical Fluid Mechanics and Multidisciplinary Design* 89 (2005), S. 27–44

[142] A. Jameson, W. Schmidt, and E. Turkel: Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes. In: *14th AIAA Fluid and Plasma Dynamics Conference*, 1981

[143] T. Schwarz: *Ein blockstrukturiertes Verfahren zur Simulation der Umströmung komplexer Konfigurationen*, Institut für Aerodynamik und Strömungstechnik, DLR, Braunschweig, Dissertation, 2005

[144] M. Allongue, and J.P. Drevet: New Rotor Test Rig In The Large Modane Wind Tunnel. In: *15th European Rotorcraft Forum*, 1989

[145] P. Beaumier, E. Chelli, and K. Pahlke: Navier-Stokes Prediction of Helicopter Rotor Performance in Hover Including Aeroelastic Effects. In: *56th Annual Forum of the American Helicopters Society*, 2000

[146] K. Pahlke, and B. G. van der Wall: Chimera simulations of multibladed rotors in high-speed forward flight with weak fluid-structure-coupling. In: *Aerospace Science and Technology* 9 (2005), S. 379–389

[147] J. Eddy: Effective Generation of Paretos Sets using Genetic Programming. In: *ASME 2001 Design Engineering Technical Conferences*, 2001

[148] B. M. Adams, L. E. Bauman, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, P. D. Hough, K. T. Hu, J. D. Jakeman, L. P. Swiler, Stephens J. A., D. M. Vigil, and T. M. Wildey: Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.0 users manual / Sandia National Laboratories. 2014. – Forschungsbericht

[149] Undertaking, Clean Sky Joint: *The Clean Sky JTI (Joint Technology Initiative)*. `http://www.cleansky.eu/content/page/green-rotorcraft`. Version: 2012

[150] S. Huber, S. Spurway, S. Link, A. DAlascio, A. Verna, M. Barber, M. L. Grojo, A. Gierczynski, L. Riviello, V. Motta, P. Rollet, N. Remer, C. Chiozzini, C. Smith, and L. Thevenot: Clean Sky / Green Rotorcraft ITD Periodic Report P7 (January 1st to December 31st, 2014) Publishable summary / European Union. 2014. – Forschungsbericht

[151] L. Piegl, and W. Tiller: *The NURBS Book (2nd Ed.)*. New York, NY, USA : Springer-Verlag New York, Inc., 1997. – ISBN 3–540–61545–8

[152] C. Stanger: *Erweiterung des Prozesses zur Rotorblattoptimierung um automatische Anpassungen der Strukturmodellierung*, Institut für Aerodynamik und Gasdynamik der Universität Stuttgart, Mastersthesis, 2012

[153] C. Stanger, M. Hollands, M. Kessler, and E. Krämer: Adaptation of the Dynamic Rotor Blade Modelling in CAMRAD for Fluid-Structure Coupling within a Blade Design Process. In: *18. DGLR-Fach-Symposium der STAB*, 2012

[154] W. Yamazaki, and D. J. Mavriplis: Derivative-Enhanced Variable Fidelity Surrogate Modeling for Aerodynamic Functions. In: *AIAA Journal* 51 (2013), S. 126–137

[155] F. Neri, and V. Tirronen: Recent advances in differential evolution: a survey and experimental analysis. In: *Artificial Intelligence Rev* 33 (2010), S. 61–106

[156] C. B. Allen: CHIMERA volume grid generation within the EROS code. In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 214 (2000), 125-140

[157] N. Hariharan, M. Potsdam, and Andrew Wissink: Helicopter Rotor Aerodynamic Modeling in Hover Based on First-Principles: State-of-the-Art and Remaining Challenges. In: *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2012

[158] L. F. Richardson: The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam. In: *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 210 (1911), Nr. 459-470, S. 307–357. – ISSN 0264–3952

A Appendix

A.1 Details on Surrogate Based Optimization

The idea of this chapter is to give more insight into the various available surrogate models. Especially since Kriging is interpreted as a combination of a polynomial regression model and radial basis functions, they are explained here. At the end, the here used optimization algorithms are discussed.

A.1.1 Surrogate Models

Polynomial Regression Model

The ansatz for a function $y(\vec{x})$ is given by a polynomial ansatz $\hat{y}(\vec{x})$. For a polynomial function of second order without bi-quadratic terms, this is written as:

$$\hat{y}(\vec{x}) = \beta_0 + \sum_i \beta_{1,i} x_i + \sum_i \beta_{2,i} x_i^2 \tag{A.1}$$

Written as a vector multiplication:

$$\hat{y}(\vec{x}) = \vec{f} \cdot \vec{\beta} \tag{A.2}$$

with $\vec{f}$ the regression vector dependent on the input vector $\vec{x}$.

$$\vec{f}(\vec{x}) = (1, x_1, \dots x_k, x_1^2, \dots, x_k^2)^T \tag{A.3}$$

The coefficient vector $\vec{\beta}$ is computed by a least squares approach using the sampled responses $\vec{Y}_s$ and the pseudo inverse of the regression matrix $\mathbf{F}$

$$\vec{\beta} = \mathbf{F}^+ \vec{Y}_s = (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T \vec{Y}_s \tag{A.4}$$

which is a collection of the regression vectors $\vec{f}$ of the sample points:

$$\mathbf{F} = \begin{bmatrix} \vec{f}(\vec{x}_1)^T \\ \vdots \\ \vec{f}(\vec{x}_i)^T \\ \vdots \\ \vec{f}(\vec{x}_n)^T \end{bmatrix} \tag{A.5}$$

The advantage of the polynomial regression model is that it is easy to implement, robust, and one of the fastest methods available. However, as the name implies, the model regresses the function and data points are not necessarily re-interpolated when the number of sampling points exceeds the degrees of freedom of the polynomial model.

Radial Basis Function Approximation

Radial basis function approximation is a linear combination of radial basis functions ψ and corresponding weights w. The bases are located at the sample points and therefore the number of bases is equal to the number of samples. The approximation is given by:

$$\hat{y}(\vec{x}) = \psi(\vec{x}_1)w_1, \dots \psi(\vec{x}_n)w_n = \vec{\psi}(\vec{x}) \cdot \vec{W} \tag{A.6}$$

where $\vec{\psi}(\vec{x})$ is the correlation vector and $\vec{W}$ are the predetermined weights of these correlations. An entry of $\vec{\psi}$ is built as follows:

$$\psi_{v,q} = RBF(\sqrt{|\vec{x} - \vec{X}_{s,q}|^2}) \tag{A.7}$$

with RBF being a radial basis function, which is supplied the radial distance of the input vector $\vec{x}$ and the sample point $\vec{X}_{s,q}$ representing a basis point. The weights $\vec{W}$ are computed using the inverse of the correlation matrix $\mathbf{\Psi}$ or any other suitable linear algebra solving algorithm.

$$\vec{W} = \quad \mathbf{\Psi}^{-1}\vec{Y}_s \tag{A.8}$$

$$\tag{A.9}$$

The correlation matrix $\mathbf{\Psi}$ represents the collection of correlation vectors of each point in the sampling matrix $\mathbf{X}$:

$$\Psi_{i,j} = \quad RBF(||\vec{X}_i - \vec{X}_j||_2) \tag{A.10}$$

$$\tag{A.11}$$

with the dimensions $n \times n$. The advantage of this approach is that it is non regressive, thus the responses will go through the sampled points. Radial basis functions face two problems though; On the one hand, the inversion of the correlation can become ill-posed if data points are being too close to each other or if the function is noisy. The latter occurs if unconverged simulations are used, or a true physical experiment is performed. In both cases, the numerical instability and noisiness, a coefficient Λ may be added to the diagonal of correlation matrix $\mathbf{\Psi}$ to alleviate the problem:

$$\mathbf{\Psi_{noise}} = (\mathbf{\Psi} + \Lambda\mathbf{I}) \tag{A.12}$$

making the RBF approximation slightly regressive.

A.1.2 Optimization Algorithms

Differential Evolutionary Algorithm

A popular optimization method is the evolutionary algorithm developed by Storn and Price [116] referred to as differential evolutionary (DE) algorithm. It belongs to the

class of heuristic global optimization procedures and is based upon randomized vector operations on a given population of individuals representing the samples in the design space. First a random set of design vectors is created, with at least four individuals in this set. Then, for each member, a trial vector is evaluated. One dimension of the trial vector is generated as follows:

$$\vec{v}_i = \vec{x}_t + F(\vec{x}_r - \vec{x}_s) \tag{A.13}$$

with i being the dimension modified, r, s, t arbitrary random integers all unequal to each other and different from the index of the current design vector. $F \in [0, 2]$ determines the strength of the change and is similar to the mutation rate of genetic algorithms. As $\vec{x}_i$ is initialized with an original member, only certain dimensions of this original member are mutated. The choice, which dimension is overwritten is decided by:

$$\begin{cases} \text{if (rand\_real (1)} \leq \text{CR) or j=rand\_integer (i)} & \vec{x}_{i,j}^{new} = \vec{v}_{i,j} \\ \text{else} & \vec{x}_{i,j}^{new} = \vec{x}_{i,j}^{old} \end{cases} \tag{A.14}$$

where rand_real (x) is a real random number evaluated in $[0, x]$, CR, the crossover rate, which is user specified between $[0, 1]$, rand_integer (i) an integer number between $[0, i]$ and j the current individual. The evolutionary process happens, when the trial vector $\vec{x}_j^{new}$ of each member is evaluated. In case the trial vector is better than the original vector, then the original vector is replaced, otherwise the original vector remains in the population.

The survey by Neri and Tirronen [155] shows that the 'Differential Evolutionary Global and Local' (DEGL) algorithm as mentioned in Das et al. [133] is an enhanced version over the standard Differential Evolution scheme. The idea is to locally and globally refine the population by using two different mutation operations:

$$\vec{L}_i = \vec{x}_i + \alpha(\vec{x}_{n,best} - \vec{x}_i) + \beta(\vec{x}_p - \vec{x}_q) \tag{A.15}$$

$$\vec{G}_i = \vec{x}_i + \alpha(\vec{x}_{p,best} - \vec{x}_i) + \beta(\vec{x}_r - \vec{x}_s) \tag{A.16}$$

$$\vec{v}_i = w\vec{G}_i + (1 - w)\vec{L}_i \tag{A.17}$$

L_i is the local mutation vector, with $\vec{x}_{n,best}$ being the best vector in the neighborhood, and $\vec{x}_p, \vec{x}_q$ two randomly picked vectors from the neighborhood. The neighborhood itself is defined by the indices and ranges from $i-k \ldots i+k$ with k being a user specified constant. G_i is the global mutation vector and works in a similar way as the local mutation vector. Instead of choosing members from the neighborhood, members from the whole population are selected. α and β behave in a similar way as F, and are set equal according to Neri and Tirronen [155]. The final mutation vector $\vec{v}_i$ is then composed of a weighted sum of both of $\vec{L}_i$ and $\vec{G}_i$. Das et al. [133] propose three methods for selecting the weights,

$$w_i = \frac{g}{g_{max}} \tag{A.18}$$

$$w_i = exp(\frac{g}{g_{max}} ln(2)) - 1 \tag{A.19}$$

$$w_i = w_i + \alpha(w_{best} - w_i) + \beta(w_r - w_s) \tag{A.20}$$

While the first two weight variants are determined upon the current generation g and maximum number of generations g_{max}, the weights for the latter version are adjusted for each individual and evolve with the individuals. The weights from the randomly picked vector $\vec{x}_r, \vec{x}_s$ as well as the weight of the best individual and the to be mutated individual are taken into account. The crossover is then applied as in Equation (A.14).

When successfully converged, the DE algorithm has a population where all members are more and more equal and circle in on the global optimum. An advantage over the genetic string evolution is that this algorithm works with less individuals and is not prone to gaps caused by the encoding, thus is more accurate.

The DEGL algorithm has been adjusted in this work to find Pareto fronts for multi-objective optimizations by taking advantage of the sorting algorithm NSGA-II by Deb et al. [91], as proposed for their genetic optimization algorithm.

Non-dominated Sorting Genetic Algorithm-II Operator

This section briefly describes the mechanism of the NSGA-II algorithm by Deb et al. [91], which compares two individuals in a genetic algorithm. The problem with multi-objective optimizations is that on a non-dominated Pareto front, it is difficult to say which sample is better than the other. Especially when a genetic or evolutionary algorithm tries to determine which individual survives and which dies.

The idea is to determine so called ranks of Pareto fronts. For a set of points, the Pareto optimal ones are removed from the set and assigned rank 1. The remaining set is then checked for Pareto optimality. The Pareto optimal points of the second set are removed and assigned rank 2. This is repeated until no more samples are left in the original set and each sample is assigned a rank. If the genetic algorithm is now comparing two samples with each other, the sample with the better rank wins. What happens if two samples of the same rank are compared? For this case Deb et al. introduce the crowded distance operator, which sorts the individuals in each rank by the crowded distance. The crowded distance is determined by finding the next neighbors of the current sample and computing the distance in each goal function value from the neighbors. Anchor points, which only have one neighbor, are assigned a distance of infinity in order to always preserve them. With this, the genetic algorithm picks the sample with the larger distance, as this means less samples are in the area of the current front and it is expected that more interesting samples can be found there.

For implementation purposes, it is easier to sort the whole set apriori and only do an index comparison by the genetic algorithm. First determine rank 1 and compute the distance for all points, sort them accordingly and add them to the new set. Then repeat these two steps for the remaining points of the original set. This way everything is stacked as necessary and the check for which individual is better is simplified as the lower index wins.

Pattern Searches

Pattern searches are by themselves local optimizers. By employing them in multiple regions of the problem, globality is achieved. For example a DoE can be started, and then a local search is started at each point. However, by means of a hybrid optimization strategy, they are used after a global, evolutionary algorithm has found the region of

interest to refine the optimum.

Hooke and Jeeves

Named after its inventors Hooke and Jeeves [112], this algorithm is one of the first and most rudimentary optimization algorithms out there. The idea is that from the starting point a hyper cross is spawned. The hyper cross is constructed by evaluating the goal function to the left and right of each parameter, similar to a finite difference of second order. With all points evaluated, the algorithm moves its new center to the best evaluated point and starts over again. If the center point is already the best, the step size of the hyper cross is shrunk and the evaluation repeated. This is performed until a maximum number of iterations, the step size reached a minimum threshold, or sufficiently small changes in the goal function are found.

Simplex Algorithm

The simplex algorithm is based on the deterministic approach proposed by Nelder and Mead [113]. The algorithm is starting with a simplex of arbitrary size. Then the repeating part starts:

1. Order all vertices in the simplex by their goal function value. $y(\vec{x}_1) < ... < y(\vec{x}_i) < ...y(\vec{x}_{k+1})$
2. Compute the multi-dimensional center of gravity $\vec{x}_0 = \frac{\sum^{k+1} \vec{x}_i}{k+1}$
3. Reflect a new point through the formula $\vec{x}_r = \vec{x}_0 + \alpha(\vec{x}_{k+1} - \vec{x}_0)$. If the point is better than the worst, but not better than the best, replace the worst point $\vec{x}_{k+1}$ with $\vec{x}_r$, and go back to step 1.
4. Expand a new point by $\vec{x}_e = \vec{x}_0 + \gamma(\vec{x}_{k+1} - \vec{x}_0)$, if $\vec{x}_r$ is the best point so far. If this point is better than the reflected point, replace $\vec{x}_{k+1}$ with $\vec{x}_e$, if the reflected point is better than $\vec{x}_e$, replace $\vec{x}_{k+1}$ with $\vec{x}_r$ and go back to 1. If no improvement is given, continue.
5. Contract the simplex by computing a point at $\vec{x}_c = \vec{x}_0 + \rho(\vec{x}_{k+1} - \vec{x}_0)$. If the new point $\vec{x}_c$ is better, then replace it with $\vec{x}_{k+1}$ and go to step 1
6. Reduce the simplex completely, by contracting all points except the best by $\vec{x}_i = \vec{x}_1 + \sigma(\vec{x}_i - \vec{x}_1)$

Common values for α, γ, ρ and σ are $1.0, 2.0, 0.5, 0.5$. This algorithm is fairly robust, as it does not require gradients. However, it also suffers from the problem of choosing the appropriate initial size and position of the simplex, similar to a gradient method utilizing finite differences.

A.1.3 Multi-Dimensional Slicing

The human brain can process two- and partially three dimensional spaces. While this thesis is printed on a two-dimensional plane, it is difficult to visualize a four or ten dimensional space. However, when breaking the design space down to two dimensional

projections, the design space can be presented in slices. As the idea of projections can be grasped, a common misunderstanding is that the behavior of the presented is the same, even if one of the not projected dimensions has changed.

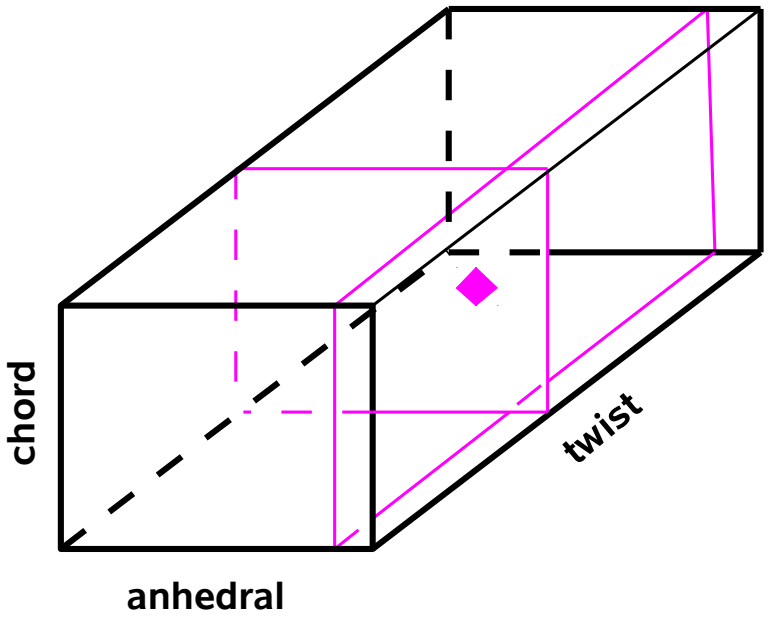


Figure A.1: Explanation of multi-dimensional slicing.

An example of slicing is given in Figure A.1. A three dimensional space is depicted with the parameters anhedral, chord and twist. When investigating the design space, an arbitrary anchor point for slicing may be chosen, which is here drawn as a magenta diamond. Now slicing the design space with orthogonal planes that all have this anchor in common, yields the projections indicated by the magenta rectangles. The chord-twist projection is intentionally not plotted for a better view. Be aware that the anhedral-chord projection may look very differently if the twist parameter is changed, which also happens to the chord-twist projection if the anhedral is changed. Another, more physical example, which is limited to three dimensions, is the flow field around an arbitrary body. Let the original parameters anhedral, chord and twist be the spatial coordinates x, y, z and the goal function be pressure p. A slice of the $x-y$ plane at $z=0$ yields a different contour plot of the pressure than at the location $z = z_{body}$. A fourth dimension may be introduced by adding time t. A flow phenomenon may appear differently at $t = t_{pulse}$ than at $t = 0$. Going back to the original optimization problem, a fourth dimension might be the blade sweep, which adds an additional complexity to the system.

A.2 Details on Aerodynamic Models

A.2.1 Comparison of Aerodynamic Models

From the theoretical discussion in Section 3.2, the various methods are categorized into low-, mid- and high-fidelity methods, listed in Table A.1. The idea is to give the reader an overview of the available physical effects being modelled by each method and to point out their restrictions.

Fidelity	Very Low			Low		Mid		High
Methods	BET	BEMT	BET & finte state inflow	BET & presc. wake model	BET & free wake	Panel Method & free wake	Euler	RANS
boundary layer/ friction	Tab ⊙-⊕	Tab ⊙-⊕	Tab ⊙-⊕	Tab ⊙-⊕	Tab ⊙-⊕	N/A ⊖	N/A ⊖	viscousity turb.model ⊕
compressible flow	Tab CF ⊙	Tab CF ⊙	Tab CF ⊙	Tab CF ⊙	Tab CF ⊙	CF ⊖	FS ⊕	FS ⊕⊕
laminar-turb. transition	Tab ⊙	Tab ⊙	Tab ⊙	Tab ⊙	Tab ⊙	N/A ⊖	N/A ⊖	N/A ⊖
stall	Tab ⊙	Tab ⊙	Tab ⊙	Tab ⊙	Tab ⊙	N/A ⊖	FS ⊖	FS ⊕
downwash	CF ⊖	momentum theory ⊙	CF VX ⊙	CF VX ⊙-⊕	CF VX ⊕	FS ⊕	FS ⊕	FS ⊕
3D effects/ vorticity	CF inflow models ⊖	CF ⊖	CF VX ⊖-⊙	CF VX ⊙-⊕	CF VX ⊙-⊕	VX ⊕	FS num. damping ⊕-⊕⊕	FS num. damping ⊕
Application n	Flight mechanics			Acoustics/ Performance		Acoustics Acoustics	Acoustics/ Performance	Performance/ /Loads

Table A.1: Methods as implemented in comparison with different aerodynamic effects. Legend: CF- corrections factors, FS- complete field solution, VX- velocity field, Tab- tabled data; ⊖- poor/no, ⊙- fair, ⊕- good, ⊕⊕- very good prediction capabilites.

A.2.2 Discretization Notes for 7A Rotor

Table A.2 lists the line and surface discretization of the 7A derived blades with the respective solver. Special notes have to be made for the 7A computations concerning:

- **UPM** in forward flight: A wake roll up model at the tip is used, which stabilizes the computation. The Kutta condition is enforced only by ensuring parallel flow on the Kutta-panel, not the strict pressure match.
- **EU** in hover flight: the artificial dissipation has been decreased for the central scheme. The k2 parameter is set to $\frac{1}{32}$ instead of $\frac{1}{2}$. This yields blade configurations closer to **FNS** at the cost of a few rotors failing, in particular those where shocks occur.
- **EU** in forward flight: these are single blade computations. $1\frac{1}{4}$ revolutions are computed for each coupling step.
- **NS** in forward flight: these are single blade computations. 2 revolutions are computed for each coupling step.
- **FNS** in forward flight: these are four-bladed computations. $\frac{1}{2}$ revolutions are computed for each coupling step corresponding to two periods in the 4/rev spectrum.

case	**hover**			**forward flight**		
locations	radial	chord	azimuth	radial	chord	azimuth
BET	31	1	10	31	1	10
BEMT	31		6			
FISUW	31		6	31		6
P.WAKE	25	1	6	25	1	6
F.WAKE	25	1	10	2	1	10
UPM	15	31	11.25	15	31	5
EU	25	61		33	96	1
	76,032cells			147,456 cells		
NS	25	61		33	96	1
	126,126 cells			245,760 cells		
FNS	61	201		65	193	0.5
	1,376,256 cells			10,101,760 cells		

Table A.2: Discretization of the blade for the individual solvers and flight conditions for 7A configurations. Azimuth refers to the temporal resolution.

A.2.3 Discretization Notes for GRC Rotor

The discretization for the GRC blade is listed in Table A.3, with the special notes of:

- **EU**: The k4 parameter of the Jameson scheme is set to $\frac{1}{128}$ to reduce the numerical dissipation in the field. The forward flight computations are single-bladed and run $1\frac{1}{4}$ revolutions for all coupling steps.

- FNS in hover flight: The k4 parameter of the Jameson scheme is set to $\frac{1}{96}$ to reduce the numerical dissipation in the field.
- FNS in forward flight: These are five-bladed computations. The first coupling step runs a full revolution, and it is reduced by $72°$ per coupling step until $144°$ azimuthal positions remain. This corresponds to two periods in the 5/rev spectrum.

case	hover			forward flight		
locations	radial	chord	azimuth	radial	chord	azimuth
FISUW	31		10			
P.WAKE				31		5
EU	37	57		33	57	1
	110,592 cells			129,024 cells		
MEU				33	57	1
				2,242,560 cells		
FNS	73	185		65	113	0.5
	1,440,768 cells			13,025,280 cells		

Table A.3: Discretization of the blade for the individual solvers and flight conditions for GRC configurations. Azimuth refers to the temporal resolution.

A.2.4 Grid convergence of the GRC rotor meshes

While for the 7A rotor the mesh setup is taken from Imiela [125], who compares them at different grid densities, the GRC rotor meshes are made with a self-developed grid generator. It is based on techniques found in the GEROS [156] grid generator with a few extensions for improved domain decomposition for parallelization of the CFD simulation as well as enhanced geometry modeling. Now these meshes include the blade tab and the blade root and tip are modeled by rounded caps, instead of simply tapering the meshes to zero thickness. A visual comparison of the mesh topology is displayed in Figure A.2.

A grid convergence study is undertaken for the Navier-Stokes meshes of the GRC rotor made for hover and forward flight. Here, three levels are investigated, coarse, medium, and fine. The finest level is generated first, and the medium mesh only contains every other grid point and the coarse mesh every fourth grid point of the fine mesh. These mesh studies are done using the full simulation process including structural coupling and stopping criteria as these effects are also eminent during the optimizations.

In particular in hover, the residual as well as the results tend to oscillate. Likely the steady assumption is only valid up to a certain point before vortex pairing and vortex boundary layer effects start to become unsteady. Hariharan et al. [157] elaborate in detail on the difficulties of hover simulations and what research is required to eventually alleviate these problems. As none of their unsteady mentioned methods has turn-around times of the steady solution of the RANS equations in a periodic mesh, an engineering approach solves this problem. The results of the last 1000-2500 iterations are stored. They are first smoothed to damp the higher harmonic oscillations and then the mean of the left over sine wave is abstracted. Figure A.3 shows exemplary

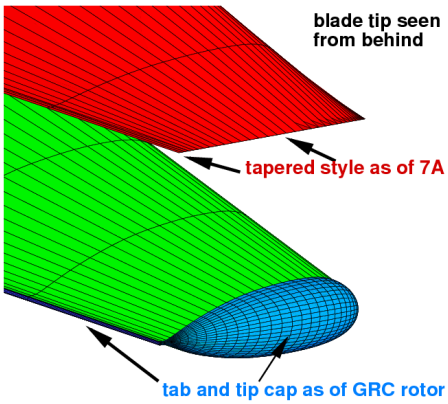


Figure A.2: Changes made from the topology Imiela [125] employed. Red is the conventional configuration, in green the new one with the novelties marked in blue.

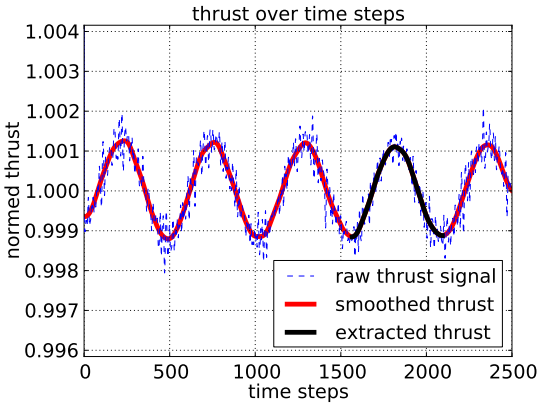
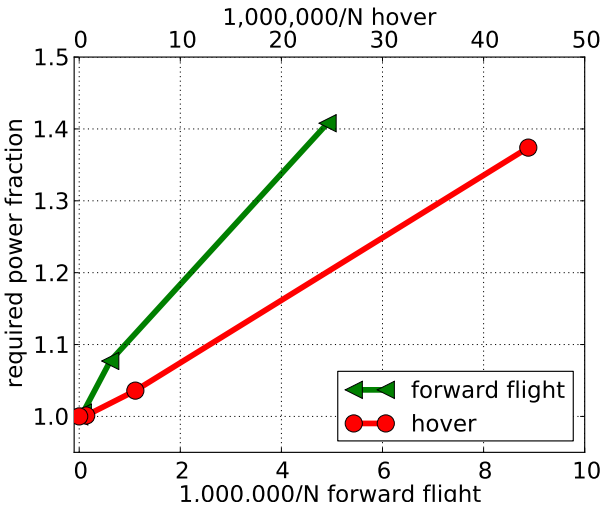


Figure A.3: Smoothing an abstraction of the hover results.

this process for the rotor thrust. While the higher harmonic oscillations are smaller when performing the simulations with a lower CFL number, the larger wave cannot be damped by this approach. From unsteady investigations it is found out that the period of this wave is over roughly three rotor revolutions and is geometry dependent. The amplitude is usually small with about $0.1 - 0.3\%$ of the averaged thrust, yet does exist. The effect increases with increasing grid density as the vortices survive longer in the simulation before they are numerically dissipated.

For this grid convergence study, the integral value of required power is directly looked upon for the trimmed rotor solution. In Figure A.4 the convergence of the integral quantities for increasing grid density is plotted with the Richardson extrapolation [158] included. The values are normed with the value of the Richardson extrapolation to illustrate the theoretical difference of the results on the individual grids with the final value. The exact numbers are listed in Table A.4.



density	cells	power
Hover		
coarse	22,000	1.374
medium	180,000	1.035
fine	1,400,000	1.001
extrapolated	∞	1.000
Forward flight		
coarse	200,000	1.408
medium	1,600,000	1.077
fine	13,000,000	1.005
extrapolated	∞	1.000

Figure A.4: Grid convergence of GRC rotor **Table A.4:** Grid convergence study of GRC meshes. Values are normed with the value meshes in hover and forward flight of Richardson extrapolation.

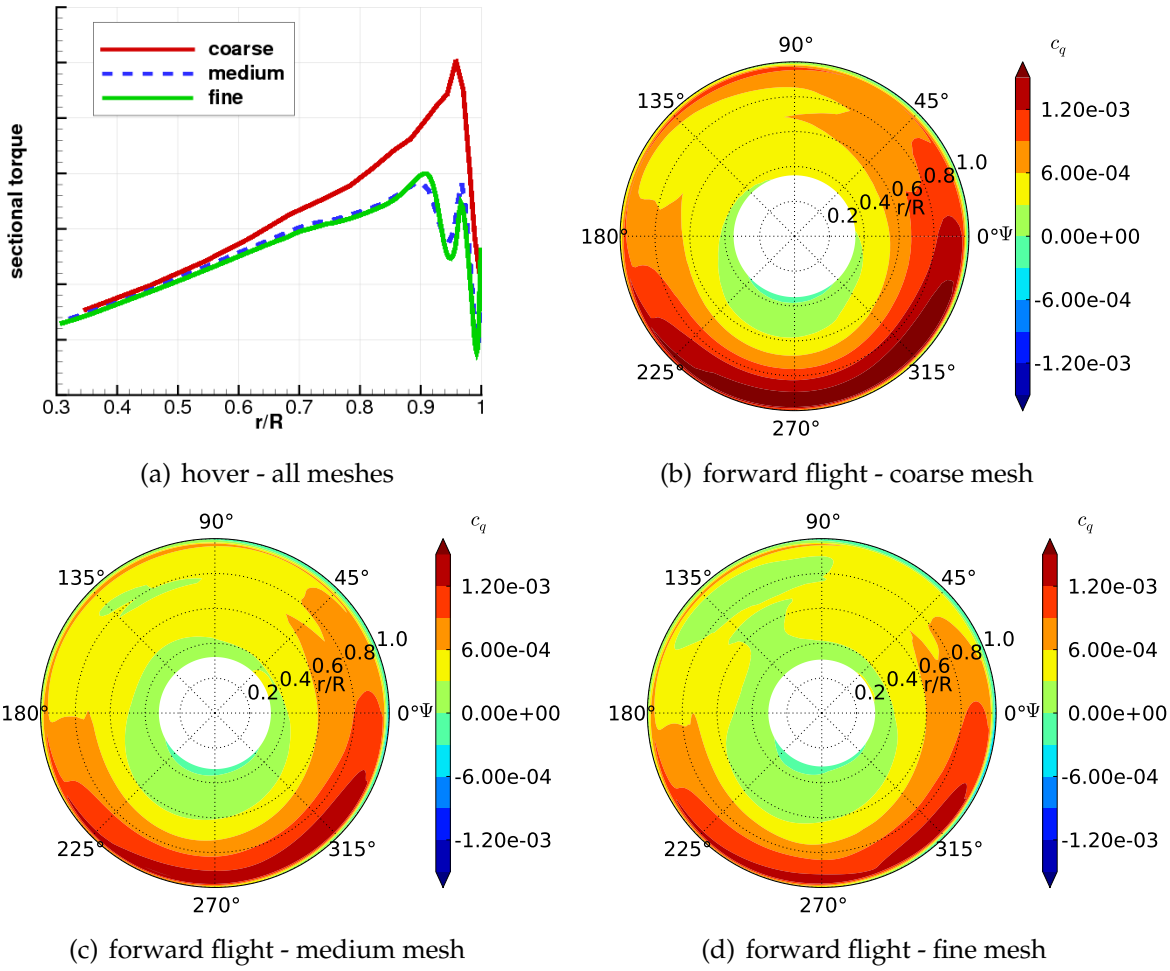
For hover, it is seen that the medium mesh has an offset of 3.6% from the extrapolated value and the fine mesh of 0.12%. As the optimizations performed in this work achieve somewhere around 6% improvement in reference to the baseline value, the fine mesh is approved to have sufficiently small changes resulting from the grid density. The sectional torque in Figure A.5(a) shows how the solution improves with the grid density. The coarse mesh barely resolves the tip vortex leading to an almost linear torque rise, while the medium mesh is close the solution of the fine mesh. It predicts the passage of the tip vortex from the previous blade similarly to the fine mesh. Also, the flow separation of the self-induced tip vortex at the blade tip is found starting at the middle grid, but only the fine grid models the very small local separation caused by the tip vortex of the previous blade accurately.

In forward flight the grid convergence is more rapidly than in hover, mostly as the Chimera setup requires more points to begin with. In Figure A.5(b) to Figure A.5(d) the torque distribution is plotted. The findings are similar to hover; on the coarse mesh a general over prediction of torque is noticed. This over prediction is strongly reduced for the medium mesh in contrast to the fine mesh, but the gap between medium and fine is smaller than coarse and medium mesh, therefore it is assumed that the grid is in the vicinity of convergence. The final deviation from the Richardson extrapolation on the finest mesh is 0.5%.

Concluding from this grid study, the deviation from the theoretical final value is sufficiently small to approve the fine meshes to the optimization problems. Another result that increases the faith in the simulation is that the overall coupling process does not interfere noticeable with the convergence of the CFD process.

A.2.5 Structural Update of the GRC Rotor

The here applied approach is developed by Stanger [152] and analyzed by Stanger et al. [153]. The idea of this approach is to come up with structural properties of a new blade geometry based upon the structural properties of a reference blade. Stanger therefore derives from structural mechanics scaling laws to adjust the reference properties to match them with the new planform shape of the blade. Assuming a cross-section of the blade is modified as sketched in Figure A.6, the stiffness changes due to the fact that the airfoil is enlarged, the mass grows, and additional levers from the offset are present. Stanger develops two routes to achieve this. One of them considers scaling the whole blade section (massive scaling), which represents a simpler process. Moments of inertia can be scaled with the square and the stiffness with the fourth power of the scaling. The other route (monocoque scaling) tries to keep the thickness of the blade skin the same and just scales the internals, such as the spar and the foam that fills the blade. If the horizontal is set equal to the vertical enlargement, and the skin thickness is neglected, the scaling simplifies to first and third order terms for the moments of inertia and the stiffnesses respectively. Stanger also proposes that these two methods may be mixed by a weighting factor. In the work presented here, the blade properties are scaled with the monocoque approach after a trial and error investigation. Table A.5 summarizes the different scaling factors for the individual properties used by HOST to model the rotor blade.



(a) hover - all meshes

(b) forward flight - coarse mesh

(c) forward flight - medium mesh

(d) forward flight - fine mesh

Figure A.5: Sectional torque of GRC rotor for all three grid sizes and both flight conditions

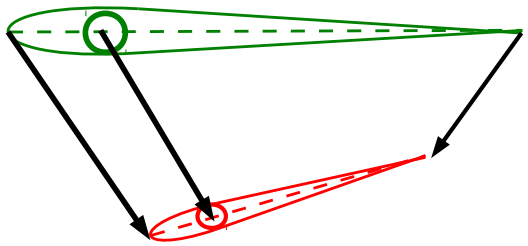


Figure A.6: Principle of sectional scaling and shifting

Property	Factor
Mass	α
Area	α
Torsional stiffness	α^3
Linear polar inertia	α
Flapping stiffness	α^3
Lead-lag stiffness	α^3

Table A.5: Scaling factors for blade properties based upon the chord enlargement $\alpha = \frac{c_{new}}{c_{ref}}$

A.3 High-Fidelity Data Generation for Industrial Scale Rotor Optimization

In order to keep the costs low, a combined sampling process for the single- and multi-objective test cases is set up; therefore computationally costly data is recycled. Figure A.7 exemplary depicts the flow of data for the individual optimizations for the multi-objective **VF Two Points Mix** case. The trend function is built from a mixed-fidelity of the low-fidelity method **P.WAKE** in forward flight and the mid-fidelity method **EU** in hover. First a design of experiments with 300 random samples is evaluated and then the mixed-fidelity surrogate models are refined through ten update cycles with ten parallel samples each. Following this, a high-fidelity design of experiments is set up by choosing the two anchor points found in the mixed-fidelity surrogate model along with 30 random samples. Then the data, here the surrogate models, are split up for the individual test scenarios. The forward flight surrogates generated from the **P.WAKE** and **FNS** method are then deviated to the single-objective optimization, while the hover surrogates from **EU** and **FNS** are channeled into the single-objective hover optimization. At the same time, all surrogates are applied to the multi-objective optimization later on.

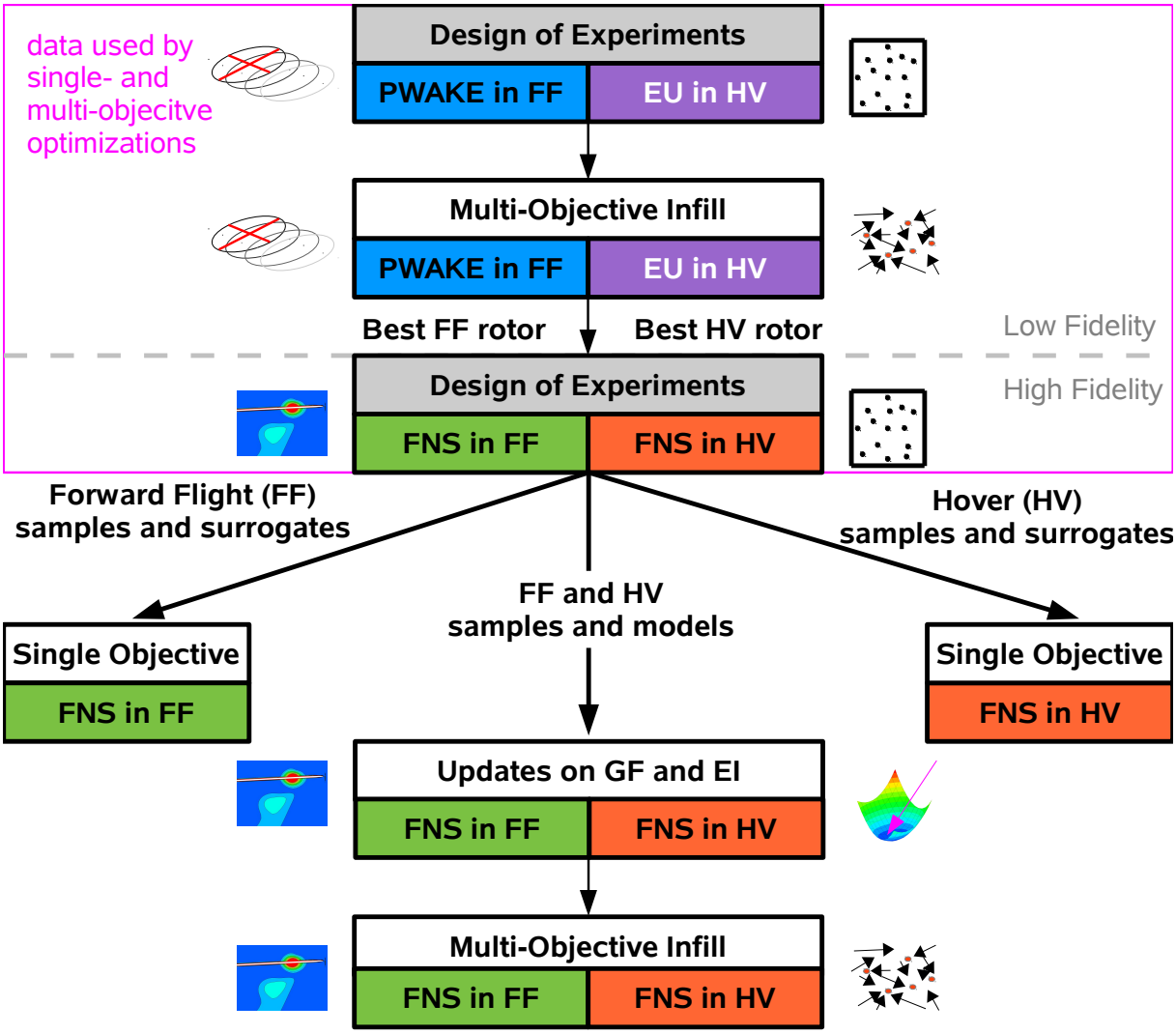


Figure A.7: Flow chart of the **VF Two Points Mix** case.